Every Month £1.25 September 1987
DENMARK DKR 40.00
GERMANY DM 9.00
NETHERLANDS DFL 9.25
SWITZERLAND SFR 6.50

SINCLAIR QL WORLD

INTERFACES
Enhancing performance

Screen storage toolkit

Compilers on test

Basic goes Super

CAMBRIDGE SYSTEMS TECHNOLOGY

24 Green Street, Stevenage, Herts SG1 3DS
Telephone: Stevenage (0438) 352150

Thor

The Thor Computer System is a professional business machine designed with the user and his future requirements in mind. Cambridge Systems Technology have developed several models making the Thor extremely versatile: the single NEC 3.5in. floppy version with or without a 20M SCSI Winchester in addition to the dual floppy model packaged in a stylish metal case. The Thor is equipped with 640K RAM, parallel and serial printer ports, battery-backed clock and a separate IBM style (PC-AT) keyboard. Supplied free with the Thor is a specially commissioned version of the award winning Psion Xchange(*) software suite and a comprehensive manual. Special features of the Thor include multitasking at a single key-stroke and enhanced screen windowing representing excellent value for money.

Thor 20

The Thor 20 Computer System is the newest development from Cambridge Systems Technology, the very latest in high-speed processing. Based on the Motorola MC68020 processor, the Thor 20 delivers on average three times the computing power of the Thor. It is available with a choice of two clock speeds: 12.5 MHz (standard) or 16.7 MHz. The Thor 20 package includes a suite of development software comprising a specially commissioned macro assembler by Talent Computer Systems and a linker by GST in addition to the Psion Xchange(*) business software together with full supporting documentation. The Thor 20 Computer System provides a substantially higher performance than the Thor at a very cost-effective price.

Thor 21

The Thor 21 Computer System is designed for 'number crunching' applications. Based on the 68020 processor and additionally incorporating the MC68881 floating point coprocessor, the performance of floating point operations are dramatically improved — taking only 1% of the time taken without the coprocessor. This system is essential for a wide range of scientific and engineering applications and only costs an additional £201.25 (inc VAT).

NEW MONITORS

CST now offer a choice of monochrome or colour monitor suitable for use with the Thor range of computers. The Microvitec 1451/DQT 14" colour monitor includes interface lead and a tilt and swivel base. The Phillips 7502 12" monitor has a dark anti-glare screen, horizontal/vertical size adjustment, position adjustment, brightness and contrast adjustment and a foldable foot. Both of these monitors have been selected by CST from the wide array of available monitors for their superior performance and reliability when used with the Thor computer range.

QDisc Interface

The best-selling floppy disk interface is fitted with a 16K EPROM containing many 'Toolkit' extensions, and CST's Ram Drive 2. It may be used with most 3.5 or 5.25 floppy disc drives. CST's own twin slimline double sided 80 track 3.5 units being exceptional value for money, with 720K of formatted storage per drive. The Toolkit provides a wide range of SuperBASIC commands and functions designed to improve access to the powerful facilities of the QL without the need for machine-code programming. Job control is made easier, files can be used for random access, alternative character sets can be produced, 'wild cards' can be used in file operations, etc.

The Ram Drive device driver allows free memory to be used as though it were a very high speed disc, in fact the fastest such device when used with the RAM-plus. Ideally used for the storage of temprary results, or multiple screen images for animated displays, it also eases the copying of files in single disc systems. The Ram Drive can only use memory which is free, so the full advantage is only felt if the QL is equipped with additional memory. Built into QDisc 4 and Thor, the Ram Drive is also available on 3.5in. and 5.25in. floppy disc.

RAM Plus

The CST RAM-plus unit expands the available memory of the QL to the limit of 640K. Using high grade 256K memory devices, this unit is the only one which offers the high performance of no wait-state operation. Housed in a rugged metal case, the RAM-plus unit has an expansion slot which duplicates the QL's, allowing any other CST peripheral to be used. Among the advantages derived from using the RAM-plus are the performance improvements of software and storage devices, and the ability to multitask several programs at once.

* Xchange is a Trade Mark of Psion Ltd.

Please supply:-	UK inc VAT		Overseas	
Qdisc Interface 4 (3.5in. util disc)	£79.95	☐	£72.70	☐
Qdisc Interface 4 (5.25in. util. disc)	£79.95	☐	£72.80	☐
QEP-III Eprom programmer (QL/THOR)	£129.95	☐	£116.00	☐
RAM-plus 512K QL memory	£139.95	☐	£125.20	☐
CST Dual 3.5in. cased floppy discs	£219.95	☐	£200.40	☐
CST Q-488 GPIB instrument interface	£224.25	☐	£198.40	☐
CST QDISC SCSI interface	£230.00	☐	£203.40	☐
Qdisc 4 + dual 3.5in drives	£275.00	☐	£249.40	☐
Qdisc 4, RAM-plus + dual 3.5in. discs	£405.00	☐	£363.60	☐
3.5in Utility disc inc. Ram Drive 2	£10.00	☐	£10.00	☐
5.25in. Utiity disc inc. Ram Drive 2	£10.00	☐	£10.00	☐
Eprom V1.16 for issue 2/3 Qdisc	£15.00	☐	£15.00	☐
Eprom V1.19 for issue 4 QDisc	£15.00	☐	£15.00	☐
Thor 1F (Single floppy)	£688.85	☐	£650.00	☐
Thor FF (Dual floppy)	£803.85	☐	£750.00	☐
Thor WF (Winchester + floppy)	£1608.85	☐	£1450.00	☐
Thor 20 1F (Single floppy)	£1292.60	☐	£1179.00	☐
Thor 20 FF (Dual floppy)	£1407.60	☐	£1279.00	☐
Thor 20 WF (Winchester + floppy)	£2212.60	☐	£1979.00	☐
Thor 21 1F (Single floppy)	£1493		£1354.00	
Thor 21 FF (Dual Floppy)	£1608.85		£1454.00	
Thor 21 WF (Winchester + Floppy)	£2413.85		£2154.00	

Available as an upgrade for existing Thor owners.
For floating point option contact CST.

For the Thor Range:

Phillips Monitor (Monochrome)	£92.95	£80.83
Microvitec Monitor (Colour) + Tilt Stand	£295.00	£256.52
Additional 12m Service contract (1F & FF)	£86.25	£75.00
Additional 12m Service Contract (WF)	£115.00	£100.00
Beige Mouse	£63.94	£57.60
Psion Chess	£24.95	£22.70

Export administration	£5.00	☐
Airmail outside Europe	£5.00	☐

Prices are inclusive of carriage
CST reserve the right to alter prices and specifications without prior notice

Please complete in full

Name .....

Address .....

.....

.....

Post Code..... Telephone .....

I enclose a cheque/PO of £ .....

Please deduct my Access/Mastercard/Eurocard/Diners Club/Vector a/c

Card No. | | | | | | | | | | | | | | | | |

Expiry Date .....

Signature .....

Access BARCLAYCARD VISA MasterCard DINERS CLUB INTERNATIONAL

SQLW9

SINCLAIR QL WORLD

Editor
Marcus Jeffery

Sub Editor
Harold Mayes MBE

Production Manager
Nick Fry

Designer
Chris Winch

Art Editors
Jim McClure
Neil Tookey

Advertising Manager
Jane Brooks Wadham

Magazine Services
Sheila Baker
Anna Morrison

Managing Editor
Brendon Gore

Publisher
Paul Coster BSc

Financial Director
Brendan McGrath

Managing Director
Richard Hease

Microdrive Exchange 089 283 4783/2952 (2 lines) TIL

Sinclair QL World
Greencoat House
Francis Street
London SW1P 1DG
Telephone 01-834 1717
Fax 01-828 0270
Telex 9419564 FOCUS G
ISSN 0268067X

Unfortunately, we are no longer able to answer enquiries made by telephone. If you have any comments or difficulties, please write to The Editor, Open Channel, Trouble Shooter, Technical Helpline or Psion Solutions. We will do our best to deal with your problem in the magazine, though we cannot guarantee individual replies.

Back issues are available from the publisher price £2 U.K. £2.75 Europe. Please telephone 089 283 4783 to check availability.

Published by Focus Magazines Ltd, London

Distributed by Quadrant Publishing Services, Sutton.

Subscription information from: TIL, PO Box 74, Paddock Wood, Tonbridge, Kent TN12 6DW.
(£15 U.K. £30 Europe. £45 rest of the world).

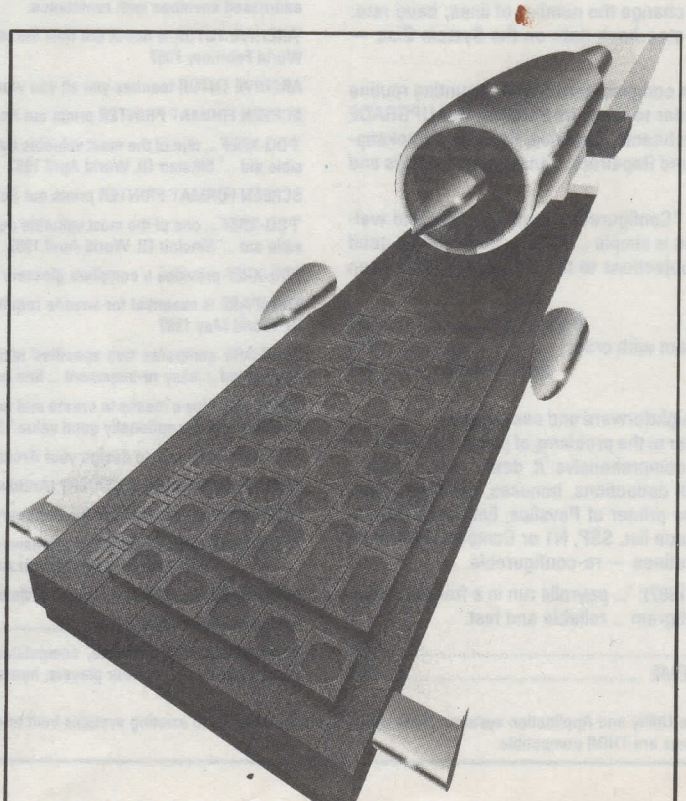
Typesetting by Jigsaw

Printing by Benham & Co Ltd

CONTENTS

SEPTEMBER 1987

7	**QL SCENE • What is happening where?**
8	**OPEN CHANNEL • Airing your views.**
10	**TROUBLE SHOOTER • More consumer problems.**
12	**THE ROMS II • Final part of our QL ROM research.**
16	**INTERFACES • Expansion possibilities.**
21	**EDUCATION • Shopping for new skills.**
24	**SUPERBASIC • First in our new series.**
28	**QL CONNEXIONS • Board? The last look.**
32	**UTILITY FILE • Our monthly utility round-up.**
37	**DIY TOOLKIT • Screen copying made easy.**
40	**COMPILERS • The QL go-faster stripes.**
45	**PSION SOLUTIONS • Solving Psion problems.**
48	**THE PROGS • The Spanish Las Vegas.**
54	**MICRODRIVE EXCHANGE • More and more every month.**



NEXT MONTH

The Electric Computer

Do you ever get fed up with constantly re-setting the clock? All you need is a little battery back-up. Perhaps your main problem is power glitches, possibly total power failure at the critical moment. There is a solution. We examine your electrical needs.

A chequered future

Next month we publish the first part of our look at intelligent game programming. To demonstrate the techniques involved the articles will develop a complete program to play the game of Checkers.

Micro Anvika

220a Tottenham Court Road, London, W1P 9AF. Tel: (01) 636 2547/580 6382
Kamla Electronics, 251 Tottenham Court Road, London, W1P 9AF. Tel: (01) 255 1966
Telex 94016414 ANVI. G (Quote 19010170) (24 hr Mail order)

Leisure

Wanderer (Pyramide)	£ 17.00
Othello (Pyramide)	£ 13.00
Vroom (Pyramide)	£ 13.00
Mortville Manor (Pyramide)	£ 17.00
Dragon Hold (Rubicon)	£ 14.00
Flight Simulator (MD)	£ 18.50
Karate (Eidersoft)	£ 13.00
Lost Pharoahs	£ 13.00
Jungle Eddi	£ 13.00
Nemesis	£ 13.00
Mr Smith	£ 13.00
Stone Raider	£ 18.00

Special Pack
Citadel/BJ 3D/Zapper
£14.95

Graphics

The Designer	£ 17.00
QL Peintre (Pyramide)	£ 13.00
Nucleon (Pyramide)	£ 17.00
Tasprint (Tasman)	£ 19.00
Tascopy (Tasman)	£ 14.00
Eye-Q (V2)	£ 29.00
Viewpoint (Rubicon)	£ 17.00
Sideways (Talent)	£ 18.00

Languages & Utilities

The Bags of tricks	£ 12.00
Digital Supercharge V1.18	£ 49.00
Digital Forth+Reversi	£ 26.00
Digital Editor	£ 24.00
Metacomco Latice C	£ 79.00
Metacomco APL	£ 85.00
Metacomco MCC Pascal New	£ 75.00
Metacomco Lisp	£ 51.00
Metacomco BCPL	£ 51.00
Metacomco Assembler	£ 33.00
Prospero Fortran 77	£ 95.00
Prospero Pascal	£ 95.00
Hisoft Devpac	£ 39.95
Talent Assembler Workbench	£ 24.00
ICE	£ 26.00
Choice	£ 15.00
Icicle	£ 18.00
QSpell (Eidersoft)	£ 24.00
Spellbound	£ 29.00
Super Media Manager (DP)	£ 36.50
Cartridge Doctor	£ 13.00
Digital Professional Astrologer	£ 57.00
Task Master	£ 24.95
Front Page	£ 21.00

Hardware

Miracle systems Trump card 760K Ram disc interface, Toolkit II and other features £199.00
Including special offer: Miracle Systems Centronics Interface free of charge.

QL Computer	£ P.O.A.
Philips CM 8833 new colour monitor	£285.00
T.V. tuner available for this monitor	£69.00

Philips CM 8533 Colour monitor
Composite Video (phono)
RGB to Linear input (SCART)
RGB/RGBI to TTL (for QL)

plus many other features	£275.00
Philips BM7502 Mono Man	£ 84.00

SEE YOU AT ZX MICROFAIR AND AT PCW SHOW-STAND NO 1815

Printers

Panasonic KX1081	£199.00
Star NL10	£239.00
Epson LX800	£229.00
Citizen 120D	£195.00
Citizen LSP10	£219.00
Miracle Centro I/Face	£ 19.00
Miracle 512K Expanderam	£ 98.00
Miracle Ram Disc Software	£ 12.00
Miracle/Q-code Modaptor Plus	£ 38.00
Miracle 1200/75 Modem	£ 49.00
Miracle WS2000 Modem	£ 99.00
Miracle Systems QL Modem	£ 49.00
Cumana Disc Interface	£ 89.00
Cumana Dual 3.5" Disc Drive including I/F	£289.00
CST Disc interface	£ 79.95
Dust Cover for QL	£ 4.95
Cart (Hard) Storage Box	£ 4.95
QL Membrane	£ 9.95
QL Transformer	£ 24.00

Psion Organiser

Psion Organiser CM	£ 99.50
Psion Organiser XP	£139.50
8K Datapak	£ 12.95
16K Datapak	£ 19.95
32K Datapak	£ 34.95
64K Datapak	£ 79.95
128K Datapak	£169.95
Finance Pack	£ 29.95
Maths Pack	£ 29.95
Spelling Checker	£ 29.95
RS232 I/F	£ 59.00
Link Up Adaptors	£ 12.95
Mains Adaptor	£ 12.95
Formatter	£ 44.95

We also stock other makes of computers, peripherals & software for Amstrad, Atari, Commodore (Amiga) etc. . .
All prices are **including VAT.** Postage above £20 on **software** is free.
Please add 50p to order under £20. For computer, printers and monitors please add £9.00 per item. Postal rates apply to **UK** only.
All goods subject to availability. Please phone for current stock availability.
Overseas orders are welcomed.
VISA AND ACCESS WELCOMED

VISA

Access

SQLW9

PDQL

Computer Systems and Software

UNIT 1
HEATON HOUSE
CAMDEN STREET
BIRMINGHAM B1 3BZ
021-233 3042

CASH TRADER UPGRADE

The latest version of this Classic Program has been completely rebuilt and TURBO-charged needs no key-in boot cartridge — very fast and bug-free — improved Save option; Save option; new Restore procedure — new CONFIGURATION program to allow many more Group and Analysis heads or documents; change the number of lines; baud rate; change to disc, RAM, Winchester etc.; have data on the System Disc — many other new features.

CASH TRADER UPGRADE provides a complete business Accounting routine — you don't have to be a Cash Trader to enjoy its benefits. Our UPGRADE customers include Accountants, Architects, Charities, Farmers, Shopkeepers, Engineers, Motor Showrooms and Repairers, Consultants, Doctors and many others.

Sinclair QL World said (June 1987) "Configuration routine a new and welcome feature ... Day to day operation is simple ... PDQL ... have done a good job on CASH TRADER ... the main objections to the old version have been overcome."

CASH TRADER UPGRADE — New users..... **£80**
Existing users send original CT Boot with order **£39**
CASH TRADER SUPPORT SCHEME — (per annum)..... **£15**

PDQ PAYROLL

A straightforward and easy to use answer to the problems of preparing salary and wages calculations — fully comprehensive it deals with monthly, weekly, and hourly paid staff — All deductions, bonuses, expenses dealt with — full reports to screen or any printer of Payslips, Employee record, analysis, deductions, cheques, cheque list, SSP, N1 or Company Pensions, P60/14, P35 — Save and Restore routines — re-configurable.

What Sinclair QL World said (June 1987): '... payrolls run in a fraction of the previous time ... a very attractive program ... reliable and fast.'

PDQ PAYROLL **£80**
Annual SUPPORT SCHEME **£15**

MAKE THE MOST OF YOUR QL AND THOR

for home or business use — Order from list or write for current catalogue. Prices include VAT and postage within the UK. Cheques with orders to, and full descriptive list available from, PDQL. Please state details of toolkit, extended memory and/or disc drive.

SPELLBOUND — the wonder Spelling Checker for Quill or Editor **£29.95**
now you can check existing documents with FILEBOUND only from PDQL **£35**
FILEBOUND is available separately at £5 — send blank cartridge and stamped and addressed envelope with remittance.

'ARCHIVE TUTOR is about the best introduction to the database system ...' Sinclair QL World February 1987

ARCHIVE TUTOR teaches you all you want to know **£21**

SCREEN FORMAT PRINTER prints out Screen, Co-ordinates and Colours **£7**

'PDQ-XREF ... one of the most valuable development utilities I have seen ... an indispensible aid ...' Sinclair QL World April 1987

SCREEN FORMAT PRINTER prints out Screen, Co-ordinates and Colours **£7**

'PDQ-XREF ... one of the most valuable development utilities I have seen ... an indispensible aid ...' Sinclair QL World April 1987

PDQ-XREF provides a complete glossary for your SuperBASIC programs £15

'COMPARE is essential for anyone requiring information concerning his files.' Sinclair QL World May 1987

COMPARE compares two specified text files in separate windows ... mismawtches highlighted ... easy re-alignment ... line numbers can be ignored **£10**

'SEDIT provides a means to create and edit screen format files in Archive ... a very useful utility and exceptionally good value.' Sinclair QL World May 1987

SEDIT the easy way to design your Archive screens **£15**

ARCHIVE DATABASE RECOVERY (Archive on the Mend) **£20**

LAZARUS the essential disc file Recovery program **£20**

PDQ NAME & ADDRESS comprehensive database system: Standard Archive **£18**
with Run Time Module for ArchDEV **£36**

MAILMERGE de LUXE the Prince of Mailmerge applications **£14**

ArchDEV/RTM (Psion) **£20**

DOMINATION the addictive, compulsive Strategy Game to test your economic and military skills — up to four players, human or computers! **£9**

MANY OTHER General Utility, Archive Utility and Application systems. NEW systems or adjustments to existing systems built to order. All PDQL SuperBASIC programs are TURBOcharged. All PDQL built programs are THOR compatible.

THE POWER BEHIND THE BUTTON

TURBO "Second to none . . . excellent . . . nothing can match it . . . takes Super BASIC & the QL into a new era" DESKTOP PUBLISHER "A very very impressive system . . . in a league of its own . . . will not make the coffee, but they are working on it" PROFESSIONAL ASTROLOGER "Descriptions such as superb, ultimate, excellent are barely adequate, a classic . . . the best of them all . . . the most comprehensive ever produced" SUPERCHARGE "A runaway success . . . QLiberator lacks Supercharge's finer features" MEDIA MANAGER "has every possible facility . . . a world beater" EDITOR "Superb . . . positively scintillates . . . a joy to use . . . DP's best program" SUPERFORTH "The definitive FORTH . . . excellent . . . very fast" EYE-Q "Brilliant . . . powerful . . . superb . . . monumental . . . the best graphic package . . . the last word" TURBO TOOLKIT "Excellent . . . comprehensive" SPRITE GENERATOR "Excellent . . . invaluable" SUPER ASTROLOGER "Excellent . . . strongly recommended" BETTER BASIC "Intelligent . . . excellent . . . an absolute must 5 stars" BACKGAMMON "Brilliant" DROIDZONE "Mind-numbing" BLOCKLANDS "Addictive" ARCADIA "We flipped over it"

The Best QL software from

DIGITAL PRECISION

New – Spectacular TURBO V2.0 SPECIAL EDITOR V2.0

ORDER FORM

To: **DP, 222 The Avenue, London E4 9SE** (or use the Credit Card hotline **01-527 5493**)

- ☐ DELUXE DESKTOP PUBLISHER (incl EYE-Q, SPECIAL EDITOR) ..... £124·95
- ☐ DESKTOP PUBLISHER + SPECIAL EDITOR ..... £104·95
- ☐ TURBO BASIC COMPILER SPECIAL EDITION V2.0 + TOOLKIT ..... £99·95
- ☐ DESKTOP PUBLISHER + EYE-Q ..... £94·95
- ☐ DESKTOP PUBLISHER V2.0 (EXPANDED QL) ..... £69·95
- ☐ PROFESSIONAL ASTROLOGER WITH ASTRONOMER V1.5 ..... £69·95
- ☐ PROFESSIONAL ASTROLOGER V1.5 ..... £59·95
- ☐ SUPERCHARGE V1.19 BASIC COMPILER ..... £49·95
- ☐ SUPER MEDIA MANAGER V1.12 ..... £39·95
- ☐ THE EDITOR SPECIAL EDITION V2.0 (EXPANDED QL) ..... £39·95
- ☐ SUPERFORTH WITH REVERSI V2.0 ..... £34·95
- ☐ EYE-Q (MOUSE VERSION)V2.1 ..... £34·95
- ☐ EYE-Q V2.1 GRAPHICS SYSTEM ..... £29·95
- ☐ SUPER SPRITE GENERATOR V4.0 ..... £29·95
- ☐ TURBO TOOLKIT V2.0 ..... £29·95
- ☐ THE EDITOR V1.17 ..... £29·95

- ☐ SUPER ASTROLOGER V1.8 ..... £24·95
- ☐ CARTRIDGE MEDIA MANAGER V1.12 ..... £24·95
- ☐ BETTER BASIC EXPERT SYSTEM V2.0 ..... £24·95
- ☐ SUPERDUMP (24 MODE PRINTER DUMP) ..... £19·95
- ☐ SUPER MONITOR DISASSEMBLER V3.5 ..... £18·95
- ☐ SUPER BACKGAMMON V3.0 ..... £12·95
- ☐ PROFESSIONAL ASTRONOMER V1.5 ..... £12·95
- ☐ DROIDZONE ..... £9·95
- ☐ BLOCKLANDS ..... £9·95
- ☐ REVERSI V2.0 ..... £9·95
- ☐ ARCADIA ..... £9·95
- ☐ DELUXE ATARI-/ST TYPE QL MOUSE ..... £59·95
- ☐ GIGASOUND 3 CHANNEL SOUND BOARD ..... £69·95
- ☐ TRUMPCARD 896K(!!) INTERFACE WITH RAMDISK, TOOLKIT & FREE DP SOFTWARE ..... £189·95
- ☐ AS ABOVE, WITH DUAL NEC DSDD 3½" DRIVES ..... £404·95

(£100 Trade in allowance on 512k Sandy SuperQboard – Phone for part exchange offers on other hardware)

- ☐ ALL 5 DIGITAL PRECISION GAMES ..... £39·95
- ☐ SPELLING CHECKER FOR EDITOR + QUILL ..... £39·95
- ☐ CHEAP CARTRIDGES – BATCH OF 20, BOUGHT AT THE SAME TIME AS DP SOFTWARE ..... £25·00

- UK delivery costs are all-inclusive, Europe add 5%, elsewhere 10% to cover airmail costs.
- Program upgrades can be obtained by sending us the original cartridge (NOT the packaging or documentation) plus £10 (£12 Europe, £15 elsewhere).

Name:_____ Address:_____

_____ Post Code:_____

☐ Cheque/Cash/PO ☐ Access/Mastercard ☐ Visa/Trustcard/Barclaycard for £ _____

Card Number:_____ Expiry Date:_____ ☐ Cartridge ☐ 3½" disk ☐ 5¼" disk

SQL9

MIRACLE SYSTEMS LIMITED

14 day full money back guarantee on all products
12 month warranty on all products
All prices include VAT and P&P

QL MODEM £49

★ New printout software

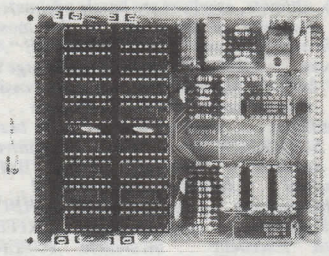
★ 1200/75 e.g. PRESTEL, T/GOLD
★ Autodial
★ 3 metre cable
★ Plugs into SER2
★ Includes Viewdata software
★ Includes 80 column software
★ Not BT approved

QL MODAPTOR £39

★ New printout software

★ Interfaces QL to standard modem
★ 300/300, 1200/75, 1200/1200
★ Autodials with DTR
★ Plugs into SER2
★ Terminated by 25 way D
★ Includes Viewdata software
★ Includes 80 column software

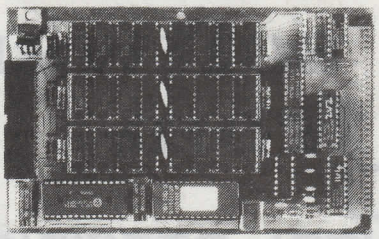
QL EXPANDERAM 512K £75



★ Increases QL memory to 640K
★ Through connector for i/f
★ Low power consumption
★ Can speed up some programs
★ Able to use larger Quill docs etc
★ Black cover included

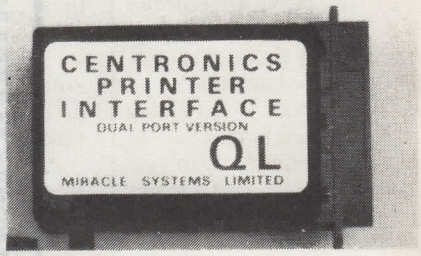
QL TRUMP CARD £199

768K RAM + DISC I/F



★ Increases QL memory to 896K
★ Standard 3½/5¼ inch disc i/f
★ Screen Dump
★ RAM Disc
★ Printer Buffer
★ Memory Cut
★ Toolkit II
★ £50 Expanderam trade-in
★ FREE 'T' shirt

QL CENTRONICS £19.50 PRINTER INTERFACE



★ Plugs into SER1 or SER2
★ Standard Centronics plug
★ Default baud rate 9600
★ 3 metre cable

QL QUADRAPRINT £19.50

★ 4 QL's to 1 printer
★ 5 metre cable £7.50
★ 10 metre cable £12.50
★ 20 metre cable £19.50

QL SERIAL CABLE £7.50

★ 25 way D plug
★ 3 metre cable

QL JOYSTICK ADAPTOR £4.99

★ Atari/Commodore J/S to QL

TRUMP CARD Plus twin 720K 3½" Disc drive £349 inclusive

Tel: (0454) 317772

Orders welcome by telephone or post

MIRACLE SYSTEMS LTD., Northavon Business Centre, Dean Road, Yate, BRISTOL, BS17 5NH, U.K.

Please send me .....

Please find cheque enclosed to the value of £.....

or debit my credit card | | | | | | | | | | | | | | | | Expiry date | | / | |

Signature .....

Send to: Name .....

Address .....

.....

.....

MIRACLE SYSTEMS LTD., Northavon Business Centre, Dean Road, Yate, BRISTOL, BS17 5NH, U.K.

QL SCENE

Powerful supply

A new battery back-up for the QL power supply has been announced by Frequency Precision. The back-up comprises a module containing nickel cadmium batteries which are continuously in circuit. In the event of a mains supply failure, power is taken from the batteries which have sufficient power to run an expanded QL for about an hour, certainly long enough to save those vital files.

In addition to providing a mains failure back-up, the system will protect your machine from mains interference. According to Frequency Precision, repeated deliberate arcing at the mains plug had no effect on the QL.

An initial batch of back-up boxes is available at £54 each, including VAT, post and packing. For further details contact Frequency Precision Ltd, Hillsview, Aller Road, Dolton, Winkleigh, Devon EX19 8QP. Tel: 08054 381.

Spellbound upgrade

One of the main criticisms of the recent *Spellbound* dictionary and real-time spelling checker from Sector Software is that, because of its nature, it cannot handle existing documents.

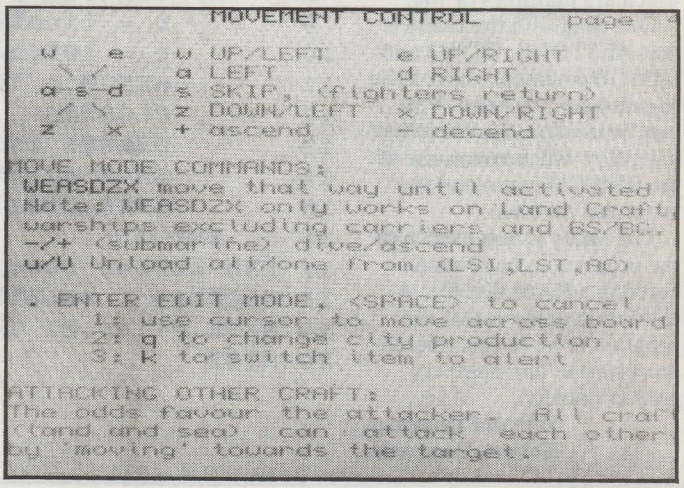
A recent add-on from PDQL, *Filebound,* looks set to change that. Filebound allows the Spellbound system to check existing Editor or Quill documents for spelling errors. Further, additions may be made to the dictionary based on those tests. As PDQL points out, it allows users with, say, a particularly scientific usage to load a few previous documents and increase the size of the dictionary quickly, rather than having to edit all those little "hat" symbols.

Filebound is available to existing Spellbound owners for £5 if they send a cartridge/disc and SAE to PDQL. New users can buy the Spellbound/ Filebound package for £35.

Untitled software

A problem managed to creep past the quality control stage in the August, 1987 Software File. Because of a missing title, the column appeared to review only the one game, *Warship*. In fact, a second review appears but without a heading, appearing to continue from the previous one. That confused many readers.

The second game is called *Domination* and was written by Anthony Trenker. It is sold by PDQL for £9.



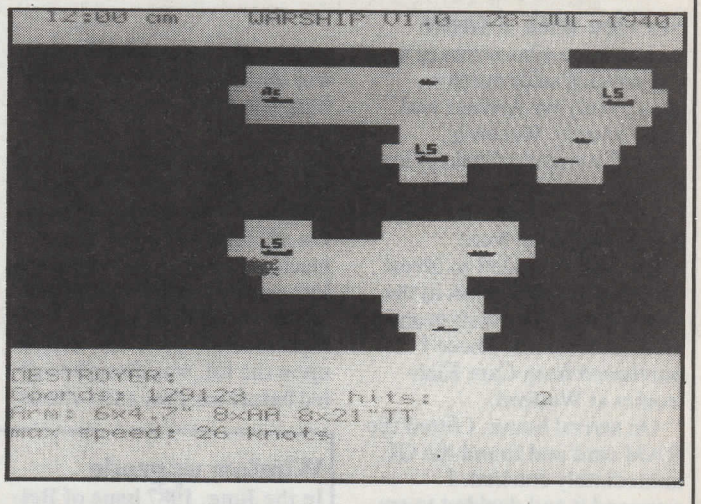
Eidersoft — the end?

Rumours have been circulating that Eidersoft may be going into liquidation. The major Eidersoft product in the QL market has been the ICE front-end system, complete with QL mouse and add-ons such as Choice, Artice and Icicle. In addition, Eidersoft has been distributing the Thor PC.

Though initially very prominent in the QL market, Eidersoft has had a low profile for many months and it seemed it may have decided to concentrate on such markets as the Atari ST and Amstrad PC. Its demise, however, would be particularly hard on the claimed 1,500 members of the Eidersoft Support Service.

Eidersoft does not see itself to be in any difficulty. Though there are indications that it has had some major problems, a solution may have emerged. It now looks as if the entire Eidersoft QL market will be sold to another party under licence with supervision from Eidersoft.

That arrangement is not yet finalised but Eidersoft claims that only a few loose ends need tying, as it has accepted the takeover offer. Under the arrangement, telephone lines will be re-directed and trading will continue under the Eidersoft name, so the change should be reasonably transparent to users.



Cash Analyser

It has always been envisaged that additions to the popular QL *Cash Trader* program would appear. One of the long-awaited additions has been an analysis routine for breaking-down and examining totals, itemising documents, distinguishing various expense entries, and so on.

A program of this kind once appeared but it was rather primitive, suffered from major shortcomings and sold only a few copies before disappearing.

A full *Cash Trader Analyser* program is available from PDQL for £25, including user guide.

OPEN

Open Channel is where you have the opportunity to voice your opinions in *Sinclair QL World*. Whether you want to ask for help with a technical problem, provide somebody with the answer, or just sound off about something which bothers you, write to: Open Channel, Sinclair QL World, Greencoat House, Francis Street, London SW1P 1DG.

Renum revised

I read with interest the article by Mike Lloyd on better editing commands in the April, 1987 issue. In that article a renumbering utility was listed — listing three — for re-numbering existing programs so that they may be listed as pages containing procedures and functions starting at line numbers which are multiples of 100.

One snag with the utility is that RESTORE, GOTO and GOSUB commands have to be re-numbered by hand. You may be interested in the following utility which makes use of the RENUM command to avoid this problem.

The utility is best used with the program to be re-numbered in a RAM disc file. A restriction in use is that the program to be re-nnumbered must start at line numbers greater than 21.

M P McClenaghan,
Hobart, Tasmania.

```
1 CLS#0 : INPUT#0;"Enter name of device and file";file$
2 MERGE file$
3 OPEN_IN#3,file$
4 REPeat read_line
5    IF EOF(#3) THEN EXIT read_line
6    INPUT#3;line$
7    gap = " " INSTR line$
8    IF line$(gap+1 TO gap+6) = "DEFine" THEN
9       b = line$(1 TO gap+1)
10      c = INT(b/100+1) * 100
11      RENUM b TO;c,2
12      CLOSE#3
13      DELETE file$
14      SAVE file$,c+1 TO
15      OPEN_IN#3,file$
16   END IF
17 END REPeat read_line
18 CLOSE#3 : DELETE file$
19 SAVE file$,21 TO : DLINE 21 TO
20 STOP
```

Service with a smile

As someone who works in a fast-moving sales environment, I rely on Archive and Quill heavily. Recently, through a stupid mistake, I lost most of my records and to ensure that the chances of that happening again were minimised I decided to invest some hard-earned cash in the QRAM front-end system and a 512K RAM card. Those I purchased from Care Electronics at Watford.

On arrival home, I fitted the RAM card and found the QL immediately crashed. I removed it and decided to try the Microdrive copy routine on the QRAM package, only to receive the dreaded "bad or changed medium" message on start-up. The next day fortunately I was able to make the short detour off the M1 to Care with both the QL and the offending packages.

The infectious good humour — "Working ones cost extra, sir" — and genuine concern soon restored my faith in the system and I was soon on my way with replacements which were tried and demonstrated before I left.

I spent an evening exploring the system and left it switched on while I went out for an hour. On my return I found the machine crashed, followed by a lock-up on re-start. Once again I returned to Care and explained the problem, whereupon the QL was disembowelled before my eyes and various pieces were taken to the workshop at the back — "Don't worry, sir, this modification rarely costs more than £300".

Some 20 minutes later I was handed the machine with the explanation that the early ULAs tended to overheat when called to do extra work such as driving the RAM card and that had caused the problem. An extra heatsink had been fitted and I was asked to see if the fault had been cured. For that no charge was made and I was lent a RAM test cartridge to try overnight.

I am pleased to report that the system is now running better than I had hoped, especially with the enhancement of Toolkit II, and the sparkle is now back in my work, so a big thank you to Care Electronics, which did not turn a drama into a crisis.

Peter Jackson
Market Harborough, Leics.

Window upgrade

In the June, 1987 issue of Better Basic, a three-line SuperBasic routine for finding the base of a window definition block was listed. As stated in the article, the routine works only if it is part of a program which opens all its channels in numerical order. Even then the routine may fail to work if there are any other jobs running in the machine.

To avoid those problems I have written a short assembly language routine to replace the SuperBasic. It also has the advantage that it returns the error "not open" if this is the case; it returns the error "bad parameter" if the channel is not a window channel and it is compatible with the *Q_Liberator* compiler.

To use the routine, assuming that you have either assembled it as mdv2_WIN_VARS_bin or used the SuperBasic loader, run:

```
10 a = RESPR(190)
20 LBYTES mdv2_WIN_VARS_bin, a
30 CALL a
40 NEW
```

The function WIN_VARS will now be added to SuperBasic and all the listings in the June, 1987 Better Basic article can be used unchanged. Note, however, that WIN_VARS(chan) remains constant until the channel is re-defined

CHANNEL

```
CH_LENCH        equ         $28

BV_CHBAS        equ         $30
BV_CHP          equ         $34
BV_RIP          equ         $58

RI_FLOAT        equ         $08
RI_ADD          equ         $0a
RI_MULT         equ         $0e

RET_FP          equ         $02

SET_UP          lea.l       FN_DEF,a1
                move.w      BP_INIT,a2
                jmp         (a2)

FN_DEF          dc.w        0
                dc.w        0
                dc.w        1

                dc.w        WIN_VARS-*
                dc.b        8
                dc.b        'WIN_VARS',0

                dc.w        0

WIN_VARS        move.w      CA_GTLIN,a2
                jsr         (a2)
                bne.s       EXIT
                moveq       #ERR_BP,d0
                cmpi.w      #1,d3
                bne.s       EXIT
                moveq       #ERR_NO,d0
                move.l      0(a6,a1.l),d1
                blt.s       EXIT
                addq.l      #4,a1
                moveq       #CH_LENCH,d2
                mulu        d1,d2
                move.l      BV_CHBAS(a6),a2
                adda.l      d2,a2
                cmp.l       BV_CHP(a6),a2
                bhs.s       EXIT
                move.l      0(a6,a2.l),a0
                moveq       #9,d0
                moveq       #0,d3
                lea         GET_ADD,a2
                trap        #3
                cmpi.b      #0,d0
                bne.s       EXIT
                move.l      d1,d4

                move.l      a1,BV_RIP(a6)
                moveq       #18,d1
                move.w      BV_CHRIX,a2
                jsr         (a2)
                move.l      BV_RIP(a6),a1
                move.w      RI_EXEC,a2
                bsr.s       DO_FLOAT
                bpl.s       NOSIGN
                moveq       #RI_ADD,d0
                bsr.s       DO_OP

NOSIGN          moveq       #RI_MULT,d0
                bsr.s       DO_OP
                bsr.s       DO_FLOAT
                bpl.s       POSNUM
                moveq       #RI_ADD,d0
                bsr.s       DO_OP
POSNUM          moveq       #RI_ADD,d0
                jsr         (a2)
                move.l      a1,BV_RIP(a6)
                moveq       #RET_FP,d4

EXIT            rts

DO_FLOAT        subq.l      #2,a1
                swap        d4
                move.w      d4,d5
                move.w      d4,0(a6,a1.l)
                moveq       #RI_FLOAT,d0
                jsr         (a2)
                tst.w       d5
                rts

DO_OP           subq.l      #6,a1
                move.w      #$813,0(a6,a1.l)
                move.l      #$10000000,2(a6,a1.l)
                jsr         (a2)
                rts

GET_ADD         move.l      a0,d1
                moveq       #0,d0
                rts
```

and so the multiple calls to WIN_VARS in some of the listings are unnecessary. It would be quicker to use the function only once and assign the value to a local variable.

Brian Hurley,
Ilford, Essex.

```
100 a=RESPR(190)
110 FOR n=0 TO 189
120       READ x
130       POKE a+n,x
140 END FOR n
150 SBYTES mdv2_WIN_VARS_bin,a,190
160 DATA 67,250,0,8,52,120,1,16,78,210
170 DATA 0,0,0,0,0,1,0,14,8,87
180 DATA 73,78,95,86,65,82,83,0,0,0
190 DATA 52,120,1,24,78,146,102,106,112,241
200 DATA 12,67,0,1,102,98,112,250,34,54
210 DATA 152,0,109,90,88,137,116,40,196,193
220 DATA 36,110,0,48,213,194,181,238,0,52
230 DATA 100,72,32,118,168,0,112,9,118,0
240 DATA 69,250,0,102,78,67,12,0,0,0
250 DATA 102,52,40,1,45,73,0,88,114,18
260 DATA 52,120,1,26,78,146,34,110,0,88
270 DATA 52,120,1,28,97,30,106,4,112,10
280 DATA 97,42,112,14,97,38,97,18,106,4
290 DATA 112,10,97,30,112,10,78,146,45,73
300 DATA 0,88,120,2,78,117,85,137,72,68
310 DATA 58,4,61,132,152,0,112,8,78,146
320 DATA 74,69,78,117,93,137,61,188,8,19
330 DATA 152,0,45,188,16,0,0,0,152,2
340 DATA 78,146,78,117,34,8,112,0,78,117
```

Meetings down under

A QL Users' Group has been established in Australia. The group is known as QL Australia and is a special interest group of the Australian Sinclair Users' Group Auspec. We have been operating since February and our main activities include a monthly meeting in Sydney, publishing QL-related articles in the Auspec newsletter and operating a QL software library.

At present we have 30 members and our numbers are growing. We believe that there are thousands of QL owners in Australia and we are eager to make contact with them. Our chairman, Vadim Kuchim, would be pleased to provide further information on our activities. He can be contacted via PO Box 729, Parramatta 2150 or by telephoning 02 633 4277.

Ron Potter,
Glenbrook, Australia.

Open University

Editor's comment: I have received a number of letters from readers intending to start Open University computer-related courses in the near future. With the new Home Computing Policy, the Open University hopes to "move the practical computing component of certain Open University courses from the present, largely study centre-based terminal facilities into the home".

As part of the plan, the Open University has distributed an Equipment Specification and a number of readers have enquired whether the QL meets the requirements. Unfortunately, as pointed out in the specification, "this essentially defines a system based on an IBM PC".

Though the QL meets the majority of the requirements, it does not run the appropriate operating system and cannot run the packages which will be used in the courses.

Not all bad

My friend and I are regular readers and, in the last few months, we have noticed the increasing number of letters referring to companies giving poor or bad service.

We would like to point out that they are not all the same. One of your advertisers, Sandy U.K., has been of tremendous help with either queries by telephone, or when we have paid a personal visit.

We would like to express our thanks to Sandy and bring to your attention the marvellous service it provides.

LM Scrivens & D N Burford,
Birmingham.

TROUBLE A PROBL

Bryan Davies attempts to solve more readers' problems in his role as the resident QL agony aunt.

If you are thinking about writing to complain about Microdrive Exchange and The Progs, there have been operational problems with them both but they are being brought under control and all orders or submissions should now be dealt with in a reasonable time.

We receive regular queries about the differences between versions of Qdos, the QL operating system. The principal differences are the bugs — different vesions have different bugs. While later versions are generally more usable, they are certainly not bug-free. If you want to know which versions have such bugs, send a stamped, self-addressed envelope to Digital Precision, referencing this article, and ask for its bug list.

The ROM versions covered are AH, JM, JS and MG; there was at least one earlier version but machines with that should all have been updated long ago. This list is likely to make sense only to users who have some programming knowledge. Advice is given on how to deal with each bug. There was an article on the subject in the August issue.

Can anyone tell me why software writers give differing values for the sizes of the same files? Generally the differences will not trouble the user but when a file has to be loaded via the RESPR and LBYTES commands, it is fairly important to get the reserved space value correct. The current (1.42) version of the extensions file for the Turbo compiler — turbo_tk_code — has 6,144 bytes (6K) reserved for it when booted-up. Using the Ice ROM to find the size of the file results in the figure 6098 being shown, whereas Taskmaster shows 6K — 6,162 bytes.

Different figures

In computer terms, the latter figure is 6,306 bytes but, if the 6K is meant in the normal arithmetical sense i.e., 6x1,000 rather than 6x1,024 the figure is 6,162. There is a difference of 208 bytes between the largest and smallest values — not much but the difference rises to a few K when program files are looked at.

I had not allowed for the increase in size of the latest version of this file and my boot routine still had 6,074 bytes reserved for the file but it seemed to load with no difficulty. I do not use most of the Turbo features, so cannot tell if all the extensions are usable.

Strikes are always with us and recent inactive periods in various sections of the Civil Service have caused poor service for some QL users overseas. One supplier sent me a Post Office sticker which had been placed on an order sent to the Antipodes; briefly, the message was "returned because of the Customs strike". Note it was not delayed but returned.

The customer was not amused and blamed the supplier. The problem apparently affected only countries outside Europe. The Post Office had strikes at sorting offices in central London during June and July.

Bernd-Rundiger Goetze reported not receiving goods ordered from Eidersoft. The latter answered that goods had been sent and the customer has been advised to check with his local Post Office in West Germany. Another package was prepared to send to Goetze but awaits delivery of a compiler program before being despatched by air mail recorded delivery. Will Goetze please let me know whether or not he has received everything ordered? If anything is still outstanding, give me details and say whether the local Post Office has been asked to check for parcels from Eidersoft.

No response

Letters from D. Marsh, N. M. Begg and B. Wright were sent to Byteback some weeks ago and no response has been received about them, despite a reminder being sent. No replies have been forthcoming from Eidersoft about letters from A. M. Mifsud and C. R. Worthington, or from WD Software — A. McLeod — or from Xenon — G. Andersson; reminders have been sent.

SHOOTER

EMSOLVED

Would the companies please answer?

Further complaints have been received about 4 Systems, Portfolio Software and Chromagraphica; in case anyone is still in doubt, those suppliers are now assumed to be defunct and we are not aware of any contact addresses for the principals of them.

Attempts to contact Printerland have failed and that supplier also has to be classified as "not around any more". A reader advised me that the premises of the supplier were cleared some months ago, "even down to the carpets."

Neither Miracle Systems nor Sandy commented on the problems experienced by J. Pemberton-Bates after he combined the MCS interface and Q-XT640 with his QL but the answer given by Talent — see last month's *QL World* — seemed to sum up the situation, that it is not worthwhile suppliers involved getting together to sort out the problems because very few people will have bought that combination of hardware and software. Scarcely a satisfactory situation for the user but not surprising when it is obvious that two of the suppliers concerned probably feel the third is solely to blame for it.

When you suspect a supplier of bad practice, it is desirable to contact the Trading Standards Office in the locality of the supplier's premises, as soon as possible. Without sufficient ammunition, in the way of specific complaints, trading standards officers can do nothing, even when they feel certain that a company is acting unlawfully, so give them the information on which to act.

The difficulty with almost all complaints received by *QL World* about non-delivery of goods is that the suppliers have already disappeared by the time customers complain. If you do not know how to contact your local Trading Standards Office, an address and telephone number are given at the end of this article from which it should be possible to obtain details of your local office. I have written to the local offices about three of the suppliers mentioned to see if there is more information available about them.

Complete set

Thanks are due to C. D. Ross for sending an apparently complete set of documentation for the Oberon Omni Reader. It will enable me to make a more useful response to any future queries on this device. Ross' Reader cost him £5. If there are more around at that price, I would like one.

GAP Software is a supplier mentioned in only one letter of complaint and as this supplier is in some ways typical of QL companies a few words about it may be appropriate. Peter Chambers is a "refugee" from the normal business world who seems dedicated to being helpful to his customers because of his experiences trying to use office computer systems.

Trying to run a company and write programs is not an easy task and something is bound to have second priority in that situation. With the obvious need to upgrade *Front Page*, especially once *Desktop Publisher* had been announced, some recent orders have not been despatched as quickly as they should have been. In addition, the local postal sorting centre seems to have been suffering from a few doses of "sympathetic action" — inaction, that is. Chambers and his wife are now getting the situation under control and the program upgrade should be completed by the time you read this.

Deserving case

While it is reasonable to complain about slow delivery of items ordered, I think we should accept that some programmers need encouraging and the first person to put a desk-top program on the QL market deserves encouragement. The latest version of Front Page has several additional features and machines with extra memory are catered for.

Please provide details regarding your problem when writing to complain about hardware or software. There is little I can do with comments like "It wouldn't run". Indicate what you did, what it did, what alternative actions you took. Having just replied to a letter complaining that two programs, including replacement copies of one of them, from one supplier did not work, I can only say that there was little I could sensibly have said but for the fact that the master cartridges were sent with the letter and I was able to check them.

By chance, my system is similar to that of the reader concerned. In general, that would not be the case. For the record, one cartridge worked on my system, the other did not. Whether the second one was supplied defective, or became that way when the reader attempted to use it, one could not say.

It is interesting occasionally to be able to check the item complained of but it should be appreciated that the time taken by such investigations precludes being able to deal with more than the odd one or two. If any item has to be returned, always enclose suitable return packaging and postage.

Trading Standards Offices — Central Administration.
ITSA, 3rd floor, Metropolitan House, 37 Victoria Avenue, Southend-on-Sea, Essex SS2 6DA.

Beating the bugs

Simon Goodwin concludes his survey of QL system bugs with a summary of the faults shared by every QL, regardless of vintage or nationality.

Forty-six errors in early versions of the QL operating system were described last month and I explained how they could be fixed by ROM upgrades. Now I list another 31 bugs which crop up in every version of the QL and tell you how to circumvent them. If you have found others, please let *QL World* know.

There is no easy way to define a bug. It is well-known in the computer industry that one person's bug is usually someone else's feature. It is almost as well-known that the person with the bug will be a customer and the person with the feature will be a marketeer.

In the absence of an industry-standard definition of a bug, I have set out to list all the quirks of the QL ROM which cause apparently correct programs to give unexpected results, or no results at all. I have also counted a few undocumented restrictions. Any non-violent action which prevents the entry of further commands is automatically considered a bug, unless Sinclair specifically warns against it in the big black QL User Guide.

That raises the question of how should a bug be fixed? In some cases, it is sufficient to detail the problem so that people can avoid it. If you document a bug, it is vital to tell users how to get the result they want without getting into difficulty.

Another approach is to make it impossible to reach the circumstances which cause the problem. Integer FOR loops did not work on the AH version of the QL, so Sinclair changed later versions so that you could not even type them in. That approach may be justified on grounds of expediency or efficiency but often it is just an excuse for leaving a real flaw uncorrected.

The last kind of bug-fix is a rare and wonderful thing. It lets you do what you originally wanted, in the way you intended. That is the best for customers, unless their original idea was a daft one, but it is the most expensive for all concerned. It often leads to the introduction of new problems, because a technique which used to work is clobbered by the fix, or because the correction passes you to more faulty code.

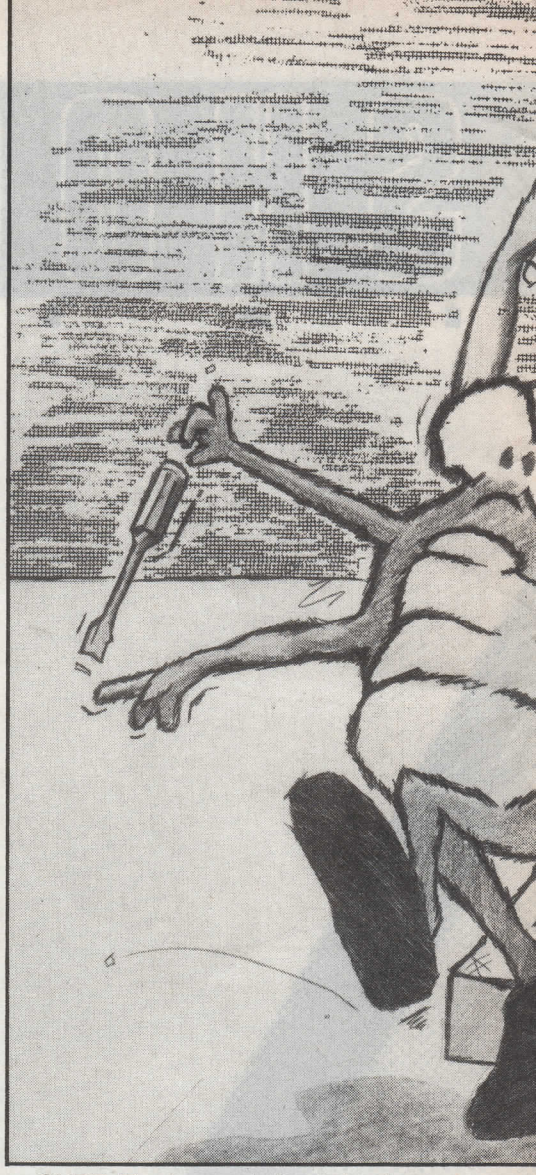
This list deals only with idiosyncracies of the QL ROM, the SuperBasic language and the underlying collection of operating routines called Qdos. Some of these bugs can cause other programs to fail. Software developers should guard against the most common problems by defensive programming in their own code, which is why I have included plenty of detailed technical information.

The problems are collected under two headings. Input/Output bugs affect the flow of information between the QL and peripherals such as drives or the display. SuperBasic bugs affect the executive of QL Basic programs; many are corrected if you compile your programs.

SuperBasic bugs

PROBLEM 1: When a REPeat or FOR statement is encountered, the value of the corresponding identifier is set to zero. For instance, this line prints five values, 0 to 4, rather than 3 and 4, as you would expect in any other Basic:
X=3: FOR X=X TO 4:PRINT X
The same problem occurs if the value of X is used to compute the end-point of the loop, or the step.

REASON: The SuperBasic interpreter does not use the same format to store loop details and simple variables. Whenever a loop starts the old value is thrown away and a new, zerod storage area is allocated.

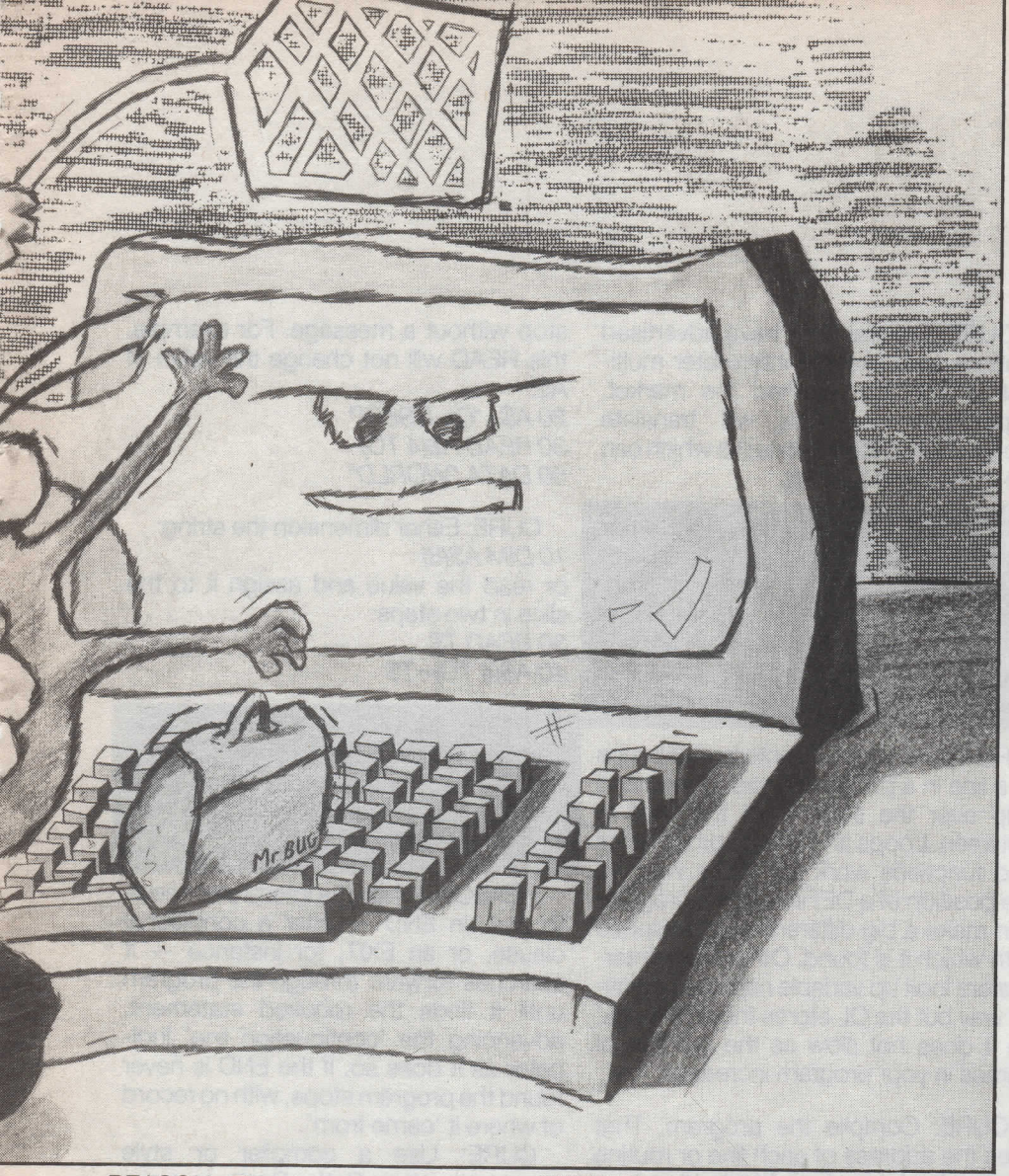


CURE: The best solution is to change the variable name — it is poor programming practice to use the same name for two logically-distinct purposes, in this case as a terminal value and an instance count. Nonetheless, there are times when that may be convenient. SuperBasic compilers do not have this bug, as they analyse the program during compilation and allocate space for the value and loop details from the start.

PROBLEM 2: The QL trigonometric package gives silly results for COS between 16384*PI radians — about three million degrees — and 65535 radians. Greater values give an overflow error. For a weird result, type:
PRINT COS(60000)

CURE: use SIN(X+P1/2) instead of COS(X), or do not do it. It is extremely unlikely that your program will fall foul of the bug unless it has gone haywire, in which case it will probably run into the overflow error. Three million degrees should be enough for anyone.

PROBLEM 3: RESPR, the function to reserve memory, does not work while a task is running. It gives a 'not complete' error.



REASON: The memory map in the Concepts section of the QL User Guide shows that the area of memory used by RESPR fits between the top of RAM and the first task loaded. Tasks may not be moved, so you cannot expand the RESPR area while any task is loaded.
SOLUTION: Use a toolkit function like ALLOCATION or ALCHP to obtain memory from the common heap at the other end of the memory map. Compilers re-direct RESPR calls to the heap automatically, since all compiled programs run as tasks.

PROBLEM 4: Every time you type RUN after a SuperBasic error — or BREAK — the system loses track of another 16 bytes of memory. You do not get this memory back until you type NEW.

REASON: The system does not tidy data allocated on the user A7 stack when an error occurs after RUN.

CURE: Start your program with GO TO instead of RUN. CLEAR and the 'Out of Memory' condition do not release the space — you must type NEW. Remember to SAVE your program first. Luckily the creepage is only small and you have to type RUN many times before you exhaust the capacious QL memory.

PROBLEM 5: You cannot de-allocate space reserved from SuperBasic with RESPR. There is no command to do this in Basic. The required machine code routine exists in the ROM — TRAP 1 with DO=15 — but it does not work; it may do nothing, allocate MORE space, or crash the system.

REASON: Resident procedure space is intended for device drivers and commands which are linked permanently into SuperBasic. If such code was overwritten the machine would crash as soon as the system tried to use it. That does not explain why the ROM routine was partly-written in the first place and it does not help people who use RESPR memory to store data such as character sets, or machine code to be accessed with CALL.

CURE: Use toolkit functions to work with space on the common heap. ALLOCATION or ALCHP will grab space and DEALLOCATE or RECHP will release it. Memory allocated by a compiled task is usually released automatically when the task stops but you can prevent this, if need be, with Turbo Toolkit. Extension commands and devices should always be linked from SuperBasic with RESPR.

PROBLEM 6: A GO SUB in a single-line FOR loop acts like an END FOR. This prints all the values of X from zero to ten, but only displays 'Counting' after the last value:

```
5 FOR x=1 TO 10: PRINT x: GO SUB 20
10 PRINT "Finished": STOP
20 PRINT "Counting": RETurn
```

CURE: Compile the program or use a multi-line FOR loop, with an explicit END FOR, instead of the short form. The problem crops up only if the GO SUB is on the same line as the FOR. Alternatively, replace the GO SUB with a procedure or function call.

PROBLEM 7: It is impossible to use an integer or string variable as the identifier of a FOR loop. Early QLs let you enter these statements but could not run them; versions from JM onwards reject the loop as a 'bad line'.

CURE: All QL compilers support integer FOR loops, although you may have difficulty entering them. Turbo and *Q-Liberator* have implicit type directions so that you can tell the compiler to make a variable an integer without upsetting the interpreter by putting a percent sign after the name. No-one has yet implemented string FOR loops.

PROBLEM 8: The integer division operator DIV does not check for all cases of binary overflow; it gives spurious results instead. Try:

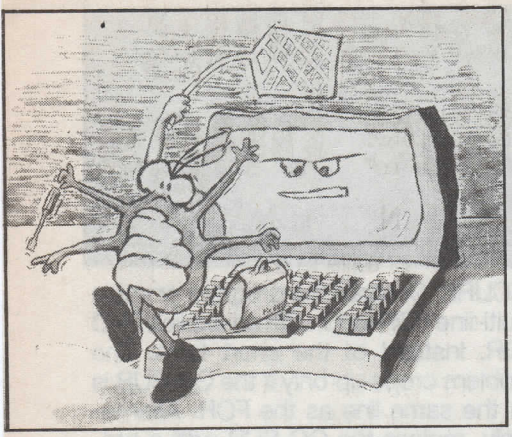
```
PRINT -32768 DIV -1
```

REASON: The QL uses the common "twos complement" format to encode signed integers into 16 bits. In this scheme, the representation of 32768 and −32768 is identical. To avoid ambiguity, the maximum valid integer is 32767 and the minimum is −32768. DIV does not check for the case when the minimum is negated; this should cause an overflow.

CURE: Avoid extreme values with MOD and DIV, or use a compiler to check your code. Turbo and *Supercharge* diagnose the overflow correctly. Q-Liberator repeats the division using floating point maths, postponing diagnosis until the value is assigned to an integer variable.

PROBLEM 9: You cannot use an integer or string variable as the identifier of a SELect statement. Early QLs let you enter these statements but could not run them; versions from JS onwards reject the statement as a 'bad line'.

CURE: Compile your program with Turbo on any QL version, or with Supercharge on AH or JM versions. Integer SELect is extremely fast.



PROBLEM 10: The INT function gives an overflow error if its argument exceeds 2 to the power of 31 minus 2, about 2.14 billion (U.S.).

REASON: The QL floating point maths scheme uses 31 bits to store the digits of a value. That allows any number from about minus 2.14 to plus 2.14 billion to be stored exactly. If a value is outside this range the computer can store it only approximately, as the first nine digits with an exponent to indicate the position of the decimal point. The machine cannot be sure of the value of the last digits. INT fails rather than return an approximate result.

CURE: Avoid such values, or use a range check to filter out values which INT cannot truncate. There is no way to INT larger values accurately without increasing the precision of the floating point format. That would require a major ROM re-write and make floating point maths much slower and more cumbersome.

PROBLEM 11: You cannot BREAK into a single-line recursive procedure.

REASON: Sinclair did not put a check for BREAK in the procedure-call code. This oversight saves time and is unlikely to cause problems.

CURE: Split the procedure over several program lines.

PROBLEM 12: You cannot interpret more than one SuperBasic program at a time.

REASON: The SuperBasic task is handled specially by the ROM — it is the only task which can grow and shrink dynamically as it runs. Most of the interpreter code would multi-task satisfactorily but interpreted programs would be very slow, because of the need to move programs around memory as they collided with one another. Much information, such as resident procedure details, would have to be duplicated between copies. The BREAK mechanism would also have to be altered, as at present it will interrupt only task 0.

CURE: Several firms have advertised routines to make the interpreter multi-task; none has reached the market. SuperBasic compilers will translate programs into standard tasks which can run within fixed bounds.

PROBLEM 13: Contrary to Sinclair claims, the speed of SuperBasic interpretation declines steadily as program size increases. A jump to the end of a large program on a standard QL can take 100 times as long as a jump to a routine at the start.

REASON: When the interpreter jumps to a line in a program it has to read and skip over the start of all the lines in between. Loops and calls to procedures and functions work the same way, so the position of a DEFinition in a program can make a big difference to the speed with which it is found. Other Basic interpreters look up variable names in a similar way but the QL stores them by index, so it does not slow as the number of names in your program increases.

CURE: Compile the program. That fixes the address of each line or routine so that it can be found instantly wherever it may be in the program.

PROBLEM 14: If you type EDIT after breaking into a procedure or function, SuperBasic can present you with a 'not implemented' error and the wrong line.

CURE: Press BREAK and type the EDIT command again. Do not be tempted to edit the line you are given, or you could corrupt the whole program.

PROBLEM 15: SuperBasic locks-up if you type CLEAR or edit a line after trying to call a procedure or function which was defined at the end of the program but deleted later. This happens only if there is no line left with a number beyond that of the DEF of the deleted routine.

REASON: The interpreter gets stuck in a loop if it is asked to clear details of a PROC/FN call which does not make sense.

CURE: Do not do it, or keep a STOP or REM on line 32767 at the end of your program.

PROBLEM 16: If you try to READ or INPUT a value into a slice of a string which has not been dimensioned, the value will not be stored and Basic may stop without a message. For example, this READ will not change the value of A$:

```
20 A$="QL USER!"
30 READ A$(4 TO)
50 DATA "WORLD"
```

CURE: Either dimension the string:

```
10 DIM A$(8)
```

or read the value and assign it to the slice in two steps:

```
30 READ T$
40 A$(4 TO)=T$
```

PROBLEM 17: If an END is missing from a program, interpretation may stop with no indication of the problem or its location. Spurious ENDs are ignored, with no message.

REASON: When the interpreter needs to find an END — after a conditional clause, or an EXIT, for instance — it searches forward through the program until it finds the required statement, advancing the 'continuation line' indicator as it does so. If the END is never found the program stops, with no record of where it 'came from'.

CURE: Use a compiler or style checker, such as Better Basic, to make sure that starts and ends are matched properly. Under such circumstances the Turbo Toolkit routines TRACE and HOW COME help you to determine where a program failed.

PROBLEM 18: Memory remains allocated every time you jump out of a procedure or function without performing a proper RETurn or reaching END DEFine. The space is recovered by CLEAR.

CURE: Do not do it. In a well-designed program, every routine should have a single entry and exit point. Avoid GO TOs and arbitrary use of REPeat and END REPeat to perform jumps, e.g.:

```
REMark Very bad style
REPeat loop
  FRED
  DEFine PROCedure FRED
END REPeat loop
END DEFine
```

This will work but it is horrendous style and will consume memory at a rate of about 1K per second.

PROBLEM 19: Very large numbers take a long time to be converted for PRINTing. Try this:

```
FOR I=1 TO 10:PRINT 123456E610
```

Beating the bugs

CONTINUED

REASON: The binary to text value conversion routine works by multiplying or dividing by 10 repeatedly until it has a nine-digit integer to deal with. This is fast for common values but very slow for extreme ones.

CURE: Wait.

PROBLEM 20: The QL internal arithmetic routines are accurate to more than nine digits but only a maximum of seven digits are displayed.

CURE: Compile the program with Turbo or Supercharge, both of which show all nine digits and correct small errors in ROM floating point routines.

PROBLEM 21: You can use only one short-form FOR or REPeat statement on a SuperBasic line. If you put more than one, only the last one will work.

CURE: Add END FORs and END REPeats to convert the loops into long forms.

Input/output bugs

PROBLEM 22: FILL sometimes colours the same line twice. This causes problems only if you are using OVER -1, as the effect is to reverse the effect of the FILL on that line.

CURE: Start drawing the pattern from a point at the top or the bottom.

PROBLEM 23: If you MERGE a file of direct commands, only the first line will be read and the file will not be closed. That makes it impossible to use another tape or disc in that drive later.

CURE: Put the Supercharge/Turbo Toolkit command END CMD at the end of the command-file line. That closes the file.

PROBLEM 24: The machine may crash if a syntax error — 'bad line' — on a line without a number is read from a command line.

CURE: Do not do it. Check your command file by putting a number at the start of each line and LOADing it; potential 'bad lines' will be marked with 'MISTake'.

PROBLEM 25: Reports may be lost or the SuperBasic interpreter may be locked out if the standard SuperBasic channels 0, 1 and 2 are CLOSEd.

CURE: Do not do it — Psion take note.

PROBLEM 26: If you load a file with LBYTES after editing and saving it there is a risk that the file loaded will not include all the changes you made.

REASON: When you CLOSE a file, Qdos prepares to replace all the parts which have been changed. They are copied from temporary storage in memory 'slave blocks' into the correct place in the file. In the interests of speed, LBYTES suspends this copying and loads the file directly from the drive, without checking to see whether or not some blocks have changed recently but have not yet been written back to the medium.

CURE: Wait for the drive to stop after CLOSE before using LBYTES, or use the Super Toolkit FLUSH command.

PROBLEM 27: The Microdrive handler gives a delayed and misleading 'bad or changed medium' message if you try to store something on a tape which is write-protected.

REASON: The handler does not check whether or not a drive is write-protected when a file is opened or data is written. Luckily for the existing data, the low-level routines detect that the tape is protected and do not allow writing. The control software assumes a 'bad medium' when several attempts to write have failed.

CURE: Beware, and do not assume disaster if you get a 'bad medium' report on one of your master tapes. Machine code programmmers can tell whether or not the currently-turning tape is write-protected with IPC call 1 but this does not tell you which drive is turning. To find that, enter supervisor mode to prevent asynchronous changes and read the drive number byte at 164078 before interrogating the IPC. A proper fix should be buried deep in the code of the device driver.

PROBLEM 28: If you set a position for binary random access far beyond the capacity of a cartridge you may get a misleading 'bad or changed medium' message, instead of 'out of range'.

REASON: The Microdrive handler uses 24-bit addresses internally, limiting the length of a file to about 16.7 million bytes. Larger values corrupt other information packed into a 32-bit register with the address.

CURE: Do not be misled by the message — your tape is intact. Then lower your sights; 16 MB Microdrives are some way off yet.

PROBLEM 29: You cannot draw a block of width 512. Nothing happens if you try it.

CURE: Use CLS or two horizontally-adjacent BLOCKs.

PROBLEM 30: The priority of SuperBasic, task 0, may be set to zero, preventing further command entry.

CURE: Use Turbo Toolkit or Supercharge extensions which check for this case and exclude it.

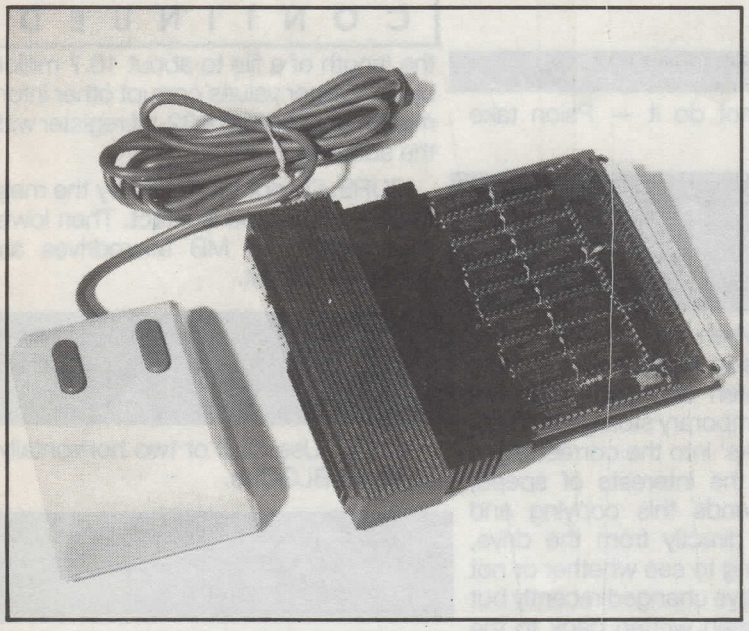
PROBLEM 31: Pressing the combination of keys CTRL, ALT and 7 — or 2 or 5 on some machines — usually causes the QL to crash at once.

REASON: These keys trigger a level 7 interrupt, which is intended to call-up a hardware debugger which Sinclair staff used while testing development systems. The facility is still there in production machines, even though the external debugger is not. The interrupt re-sets the 8049 second processor but not the main 68008. If the interrupt occurs while these two chips are communicating, as is highly likely, the 8049 'loses its place' and crashes, preventing keyboard input.

CURE: In general, do not do it. The keys have been chosen to make accidental entry very unlikely. The system call MT.TRAPV (TRAP 1, DO=7) lets you specify a routine to be executed when hardware errors or interrupts occur. Unfortunately this is of limited use, as the keyboard and RS232 die when the 8049 re-sets.

You should not treat this list as an indictment of the QL. Every computer system, of whatever vintage, contains bugs and the QL is no worse in this respect than other ambitious designs. At least, and at last, you are forewarned by this list. Each version of the QL ROM has its own bugs, besides those listed. Read the article entitled Version Therapy, in last month's *QL World,* to find more about specific ROM faults.

•BELOW: The Sandy top-of-the-range Super Q board with RAM expansion, disc interface, parallel interface, mouse port and mouse.
•RIGHT: The Infolink internal 512K RAM kit, with decoupling capacitor in the foreground.



Thanks for

Ron Massey presents a selection of add-on memory and interface devices which can be used to enhance the performance of your QL.

In the early days of the QL, the 128K RAM, fast-loading — compared to cassettes — dual built-in storage mediums, each with a capacity greater than 100K, and the 68008 processor, was all truly revolutionary. The basic QL was and still is a machine to be taken seriously.

As a bonus of genuine value, acquiring the QL also meant that new users, many of whom were new computer users as well of this great machine, also received a good word processor, database, spreadsheet and business graphics programs which, on almost any other machine reasonably comparable, would have cost more than £800 above the cost of the computer for similar programs.

In the light of continuing development, other computers have equalled and surpassed the QL native memory capacity and some include dedicated chips for sound and screen handling but, even now, few outperform it — certainly none within even its pre-Dixon price range.

It has always been the intention of manufacturers interested in the development of the QL to provide add-ons which would enhance its performance. Although early pioneers of QL peripherals such as Silicon Express, PCML and Cumana, having adopted other priorities, no longer provide their services for the QL, there is a wealth of the latest technology available.

Other companies such as Sandy, CST, Miracle, MCS and Micro Peripherals, to name a few, provide a valuable dedicated service to the QL enthusiast and are continually developing new hardware.

Fortunately, one of the conventions maintained by many manufacturers is that new hardware should remain reasonably compatible with QL configurations produced by the hardware of other manufacturers. While mixing component modules is not always a good idea, this seems a more than reasonable approach.

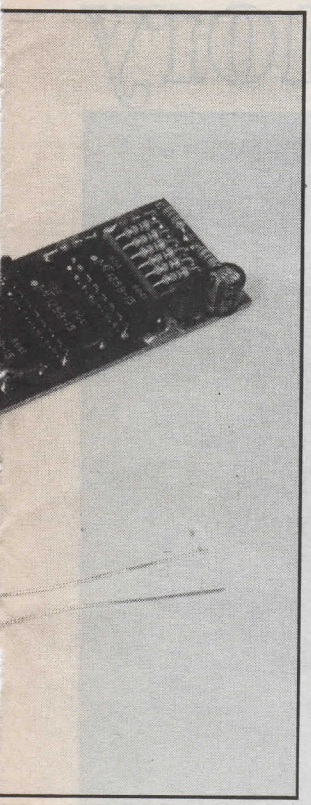
Partly for that reason, disc interfaces are available with or without memory expansion, permitting them to be used on machines which may have been converted with the 512K chip exchange upgrade offered by Silicon Express some time ago.

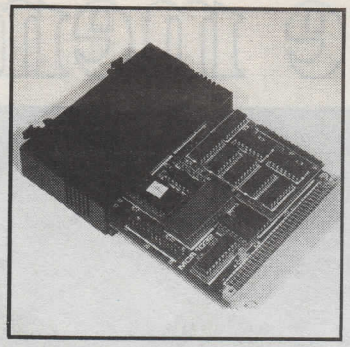
If the current range of peripherals and, by implication, future development trends are a criterion, the catalogue of new QL add-ons is well worth examination. One of the first upgrades sought by many users is to increase the QL memory. In the last 18 months there has been a dramatic fall in component prices and that has been reflected in cost to the consumer.

New horizons

Having extra memory available opens new horizons for business and home users. If for no other reason, additional memory provides elbow room to manipulate files via RAM disc or to use programs in a multi-tasking environment and enjoy the greater number of features provided in the newer, expanded-memory-only programs available for the QL.

It is not surprising that having the extra elbow room also makes many programs run faster. That is most noticeable with programs which require screen-handling as part of the operation, such as drawing programs and some types of games.

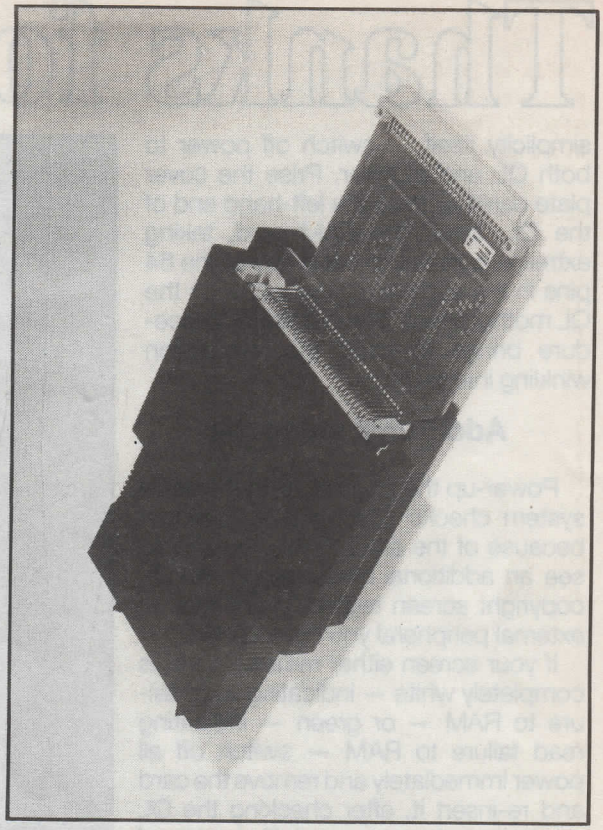


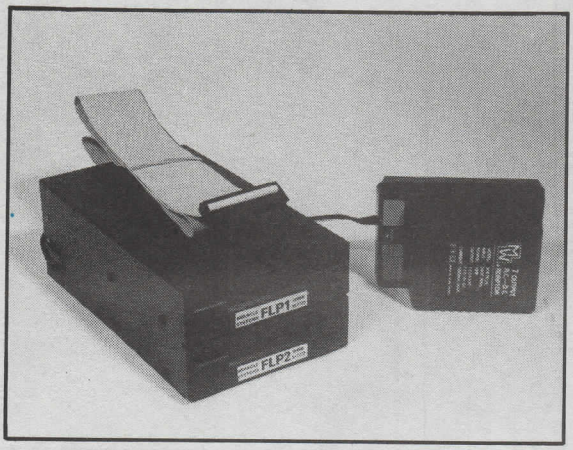


•**RIGHT: CST RAM expansion and disc interface.**

•**LEFT: MC RAM expansion and disc interface.**

•**BELOW: The Miracle dual disc drives with external power supply.**





the memory

The majority of external expansion systems incorporate a toolkit of some kind, adding new command extensions as an integral part of the QL operating system. The subject of toolkits has received in-depth coverage in previous issues of *QL World* and are highlighted only as they form an integral part of the external upgrade systems examined for this report.

Many of the early QL RAM expansion systems were available only with disc interface. Any toolkit provided was usually a cut-down version of Tony Tebby's *Toolkit II*, most of the extensions for which improved the structure of the QL file handling immeasurably.

Additional facilities enable users to emulate other QL devices. By entering "FLP_USE mdv", any file call made to Microdrive went to the disc drive, if one was available, instead. Later developments were to extend emulation to RAM disc as well.

Refinements, in the light of developing technology, enabled manufacturers to make their systems more compact and efficient. With newer, larger-capacity EPROMs, toolkits now include the complete range of commands available previously only with the software versions of the same toolkit.

Some expansion systems require that the toolkits be deliberately initialised, where they are of the Tebby/Qjump type, by inputting "TK2_EXT". Of the systems reviewed, the CST toolkit, which is hybrid Tebby/CST, is on-line from the moment the QL is switched on; the MCS interface/expansion uses its in-house toolkit.

Each of the interfaces scrutinised had features to commend them and it would be advisable to examine your particular requirements prior to buying any of those covered in this report.

Installation requirements for external RAM expansion/disc interfaces are common to all those tested and are

RAM expansion/ Drive Interface	Type	Toolkit (2) Version	Toolkit Version	Disc Driver Version	Re-set Time (sec)	Test Loop micro sec	Large Prog Load(3) (sec)	Memory remaining	(4) Free mem	(5) Free mem
External										
Sandy Super Q (1)	Tebby	2.08 R		1.17	15.84	4.882813	18.99	150K	599552	598016
CST RAM & Disc I/F	Tebby	(6)		1.19	14.25	3.662109	17.64	149K	601600	600064
Sandy Super Q & mouse	Tebby	2.10		1.18Y	14.71	4.882813	17.98	150K	599552	598016
Trump Card	Tebby	2.11 R		1.20	19.56	5.187988	19.06	405K	857600	855552
Micro Control Systems	(7)	MCS R		2.10	18.53	5.187988	19.19	151K	600576	598528
Micro Peripherals		Not available during the compilation of this report.								
Internal upgrade										
Infolink	512K add1	—	—	—	16.28	5.653783	21.49	150K	601922	—
Sandy	512K add1	—	—	—	16.01	4.999175	20.98	150K	601994	—

NOTES

1. Used as a control, this Super Q board has been in continuous use since they were first launched in May, 1986. It had the original V1.16 partial Toolkit II, upgraded in October, 1986 and now includes a built-in RAM disc.
2. An "R" indicates integral RAM disc; otherwise, RAM disc is loaded from software.
3. The large program used for this test was the following group of files: *Taskmaster, Files, Calculator, Notepad, Reset, Quill* (170K), *Quill* (100K), *The Editor* — default dataspace.
4. Memory available before accessing a drive.
5. Memory available after accessing a disc for a directory.
6. CST uses its own RAM disc; the Tebby Toolkit is a modified hybrid.
7. MCS produces its own toolkit which includes a number of useful commands. There is an overlap with the Tebby Toolkit, particularly the extensions for file handling, but other extensions are exceptional to MCS.

Thanks for the memory

simplicity itself — switch off power to both QL and monitor. Prise the cover plate carefully from the left-hand end of the QL, insert the RAM card, taking extreme care not to bend any of the 64 pins in the expansion connector on the QL motherboard. The installation procedure brings to mind the expression winkling into place.

Additional message

Power-up the QL and, after the usual system checks which will take longer because of the extra RAM, you should see an additional message on the QL copyright screen regarding the type of external peripheral you have added.

If your screen either remains dark, is completely white — indicating write failure to RAM — or green — indicating read failure to RAM — switch off all power immediately and remove the card and re-insert it, after checking the QL expansion plug to ensure that none of the pins has been bent.

Under no circumstances should any attempt be made to wiggle the card into position with the QL powered-up. To do so is likely to damage the components on the card, the QL, or both permanently.

A disc interface is a specialised device driver. Although some of the early disc interfaces specified number of tracks as either 40 or 80 and were limited to either 5-1/4in. or 3-1/2in. drive systems, modern interfaces usually support all popular disc sizes and track configurations. As a part of the disc operating system enables you can often format to a specified number of sectors. Some disc interfaces include a built-in RAM disc.

Sandy U.K. provides several alternatives for upgrading your QL for better performance to fit a variety of budgets. Its economy upgrade consists of a single 3-1/2in. 80-track Mitsubishi disc drive, a Basic interface without RAM expansion, a parallel printer port, and a range of file-handling extensions in an on-board toolkit. If RAM expansion is required for use with this interface at a later date, you can fit the internal memory card.

Disc Interface

The top-of-the-range Sandy *Super Q Board* includes its disc interface, a parallel printer port, Qjump *QRAM* utility software — the front-end system for the forthcoming *Futura*; for detailed information concerning QRAM, see the April edition of *QL World* — a completely revised mouse-driven version of *CAD-PAK*, software-based RAM disc utilities and one of the best mice available.

The ultra Super Q Board is available with or without a 512K, memory expansion card and is exceptional in a number of respects. In addition to the integral mouse port, facing to the rear of the QL, among its many design innovations it also includes a built-in mouse driver.

If bought with RAM expansion, the upgrade is supplied on two very full, separate, interconnected boards, the detachable upper card containing the 16 256K RAM chips.

The memory card connects to the board beneath it with through-pins to the main card socket and extend above the RAM PCB. I can only assume that, at some stage of development, Sandy will release hardware which will connect to those pins. If you buy RAM expansion separately, the RAM card may be added to the interface card by plugging it in as soon as you receive it.

Another important first for the QL, the Sandy Super Q no longer uses the standard 7805 voltage regulator. One of the limitations associated with these regulators, which can cause serious problems if your QL is running near its maximum capacity, is that they consume power in their own right as a requirement for their operation.

With some QLs, most notably but not necessarily the earlier releases, that could mean that the combined power drain of the QL with its add-ons can invoke the voltage regulator built-in safety feature of output "fold back" — its short-circuit protection — causing unreliable performance. In place of the standard voltage regulator, Super Q now uses a switched mode regulator, resulting in a lower power requirement and cooler running.

The latest Miracle product, *Trump Card*, has been aptly named. Not only does this upgrade give you an 896K memory but it also includes a great many very useful facilities. One of the criticisms levelled frequently at the QL is its unreliable Microdrive system. Arguments both for and against the QL native storage system seem to amount, if you have a good mdv system, to a good cost compromise for semi-permanent storage; if not, it is not worth having. Part of the argument may be due to personal

Checklist

Upgrade	Expan	Mem Price	Tebby I/F	Disc Type	Toolkit	Printer Port	features
Sandy							
Thru-con RAM card	512K	£99	–	–	–		New PCB Layout.
Internal upgrade	512K	£85	–	–	–		Optional external switch to cut memory. Fitted free.
Basic Upgrade	0K	£180	X	Partial	Par		Single-1/2in disc drive
Super Q Board	0K	£120	X	Full	Par		
Super Q Board	512K	£215	X	Full	Par		
Super Q Board	512K	£287	X	Full	Par		2-button mouse, QRAM, CADPAK
Single Disc Drive	–	£125	–	–	–		3-1/2in. Mitsubishi, cased in-house.
Dual Disc Drive	–	£190	–	–	–		3-1/2in. NEC.
Miracle							
Trump Card	768K	£199	X	Full	–		QRAM screen dumps. Memory cut, dynamic and fixed RAM discs, and printer buffer.
Disc drives	–	£189	–	–	–		Dual 3-1/2in. 80-track NEC drives. Quiet and particularly compact.
CST							
RAM expansion	512K	£139.95	–	–	–		Includes own expansion slot.
QDisk	–	£79.95	X	Full hyb	–		
Dual disc drives	–	£219.95	–	–	–		3-1/2in. NEC based.
MCS							
Memodisk	512K	£179	X	Full	Par		Printer port connection on PCB,. On-board Eigen screen dump.
INFOLINK							
Internal memory	512K	£99	–	–	–		Available from a variety of sources including Rainbow Digital Repairs, Sector Software.

Thanks for the memory

prejudice. Cartridges have two less-than-desirable features; they are more susceptible to corruption and programs usually take considerably longer to load than from discs.

Minimising both those Microdrive shortcomings considerably, Trump Card includes a feature where an entire cartridge may be imaged to either of its two RAM disc systems, dynamic or fixed. With little more than a single run of the tape — about 15 seconds — the entire contents of a cartridge are transferred to memory.

Sector map

Provided the cartridge sector map is intact, loading is reliable, even with directory curruption. Any bad sectors encountered during image loading are indicated in the RAM disc directory with "*".

Facilities for dumping screens to a printer have been included as a Trump Card feature. An adaptation of QRAM screen dump routines, one may reasonably assume that at least one of its 12 monochrome and colour printer types will be compatible with the majority of dot matrix printers supporting a graphic mode. Screen dumps are made by using the four Basic extensions:

SDUMP — May be the whole screen, a defined channel or a specified pixel area.

SDP_KEY — Used to select a "hotkey"

SDP_DEV — Assigns a screen dump device — printer port or drive.

SDP_SET — Selects one of 12 dumps, printing scale, printing to be inverse or not, random printing of red and green pixels — simulating a grey scale — or not.

Having established a recognised standard policy of pioneering product development, Miracle has made tremendous advances with its state-of-the-art QL peripherals and has added a dual 3-1/2in. NEC disc drive system in its own packaging to the range.

Considered an industry standard, NEC drives have proved to be the quietest, most reliable disc drives available. My first pair of drives, also NECs, have had considerable, totally trouble-free service for more than two years, as my collection of discs can attest. With a footprint measuring 182mm. x 116mm. Miracle drives must be the most compact unit available.

Taking a slightly different approach, CST supplies its RAM expansion and disc interface as two separate, interconnecting modules. One of the advantages of this type of design is that, if you use peripherals requiring the QL expansion port, such as the Qjump *QEP-III* EPROM programmer, you can retain your memory expansion. Except for internal memory expansion, other upgrade systems monopolise the expansion port to the exclusion of all else.

Using the latest DRAM — Dynamic RAM — chips and no marginal timings, *RAM-plus* is the fastest RAM available and may be used reliably with the CST Q+4 when two or more other peripherals are required.

One of the shortcomings of the QL is that it has only one expansion port. If you wish to use other add-ons, such as an EPROM programmer or external sound synthesisers, they must be connected to that port and the advantages gained by having memory expansion and disc drives are lost temporarily.

Several attempts have been made to provide the QL with additional expansion ports but most have met with little or no success, because of the additional separation between the QL and any memory expansion added to the system. Until recently, Silicon Express offered an internal memory upgrade by replacing the existing QL 64K RAM chips with 256K chips. This method expanded memory to a maximum of 512K and provided little increase in processing speed such as is available with faster external RAM.

Displaced chip

Two internal RAM expansion systems are available which are installed similarly by displacing the ULA chip, located next to the right of the 68008 CPU. Either one of these internal expansion boards may be installed by the end-user, although extreme care and a certain amount of expertise is required.

Once the ULA has been removed from the main QL board, the corresponding pins on the bottom of the Infolink RAM board are installed in the empty IC socket. The ULA chip is then inserted into the socket on the RAM board.

The 68008 CPU chip is removed from the main board, four of its pins are bent up and it is then re-installed. The four wires on the end of the Infolink RAM board are soldered directly to the four displaced CPU pins and, finally, the small disc capacitor is soldered to the main QL voltage regulator.

The Sandy internal memory expansion board is installed similarly except that both the ULA and the CPU are displaced for the installation and there may be only one connection required for soldering, if your QL main board is an issue 7. Otherwise, no soldering required.

A link on the Sandy internal upgrade board may be removed and replaced with two wires, terminating in an external switch for switching off the memory bank optionally. The Infolink system is available from a number of QL suppliers which, like Rainbow Digital Repairs, also offer installation. The Sandy price for its internal RAM upgrade includes installation.

Only company

Although a large number of professional applications by QL users include in-house electronic development, CST is the only company which has produced two specialised interfaces for the QL. The first is a 20MB 3-1/2in. hard disc to the QL. The second interface, the Q-488, is the official IEEE-488 standard general-purpose instrument bus and permits very high speed communication on an 8-bit parallel bus.

Future interface developments from other companies are likely to implement the newest floppy disc drive technology for 3-1/2in. drives having a capacity of between four and 12MB, rivalling the smaller hard discs and at a greatly-reduced price.

There can be little doubt that adding memory and/or disc drives to your QL is well worth considering. With the range of peripherals available, and still more being developed, there is a wide variety of equipment to suit almost every budget.

Building a system which is satisfactory for you will depend on what other peripherals you have for your QL, as much as on how and for what you use your QL.

In spite of the fact that building a computer system from modules is generally more expensive than buying a complete dedicated system, from several points of view I feel that doing so with the QL is justifiable.

Almost every type of software available for other machines costs more than similar programs of a similar quality for the QL. Of perhaps equal importance, performance of the majority of other business-orientated computers seldom equals that of the QL.

SECTOR SOFTWARE

★ 0772 454328 ★

TOUCH TYPIST

Don't even think about buying a replacement keyboard until you know how to use the one you already have. Why type slowly when you can be typing at professional speeds letters, basic programs, data etc can all be entered in a fraction of the time now taken by learning to type correctly. Did you know that it is harder to type with two fingers than it is to type correctly. Touch typist is the fastest typing tutor available for the Sinclair QL. It will teach you to type at up to 211 wpm. It is 100% machine code and has a keyboard tutorial to show you the keyboard basics, 3 teaching modes, adjustable speed and accuracy targets, very fast interactive keyboard, your results can be displayed on a graph and be saved to drive for future additions, the full lesson editor will allow you to alter and customise any or all of the 200 lessons which can then be save to drive to create a library of custom lesson sets, also included is a reward option where if selected providing you meet the requirements you specify you will be given a 100% Machine Code arcade game as a **reward for your progress Touch Typist will run from disk, microdrive or ramdisk. Available on disc or microdrive at only £12.**

THIS IS WHAT *QL WORLD* HAD TO SAY ABOUT THE PROGRAM

"Succeeding admirably as a serious course intended for beginners and experienced typists alike, Touch Typist is a well presented useful utility which includes the professional polish of good quality software"—*QL WORLD*, July 1986.

VARIOUS

Trump Card.....**£195.00** Schön pc keyboard **£99.00**
QL DUSTCOVER.....**£5.00**
MICRODRIVE LABELS ON TRACTOR FEED ..... **£3.50 per 100**
20 MICRODRIVES IN TRANSFORM STORAGE BOX **£37.00**
MIRACLE 512K EXPANDERAM WITH THROUGHPUT EXPANDS YOUR QL TO 640K **£99.00**
INTERNAL 640K MEMORY UPGRADE LEAVES YOUR EXPANSION PORT FREE **£99.00**
MIRACLE AUTODIAL MODEM.....**£45.00** ASTRACOM 1000 MODEM.....**£199.00**
MICROVITEC 1451DQ3 DEDICATED MONITOR (SUPERB GRAPHIC DISPLAY) **£260.00**

SECTOR SOFTWARE, having bought the first two Z88's from Clive Sinclair would like to contact all users of this machine for our Z88 mailing list. QL to Z88 file Transfer Software now available.

★ NEW ★

SPELL BOUND

REAL TIME spelling checker for Quill or The Editor V.1-17 onwards on the expanded Sinclair QL. Checks input **BY THE CHARACTER** against its 30,000+ word dictionary. **SWITCHABLE 5 MODES OF OPERATION** to suit a variety of typing styles with **DICTIONARY AVAILABLE AT ALL TIMES. DISPLAYS EXAMPLE WORDS ON REQUEST. NO DISCERNIBLE REDUCTION TO QUILLS SPEED OF OPERATION. DICTIONARY EXPANDABLE TO MEMORY/ MEDIA LIMITS. Available on Disc or Microdrive at only £29.95.**

QL SPARE PARTS

SERVICE MANUAL
THIS IS THE MOST UP-TO-DATE BOOK FOR REPAIRING THE QL. IT CONTAINS ALL CIRCUIT DIAGRAMS, FAULTFINDING CHARTS AND OSCILLOSCOPE DISPLAYS TO ENABLE YOU TO REPAIR YOUR OWN QL OR UNDERSTAND HOW IT WORKS **£25.00**
QL TEST SOFTWARE + RS232 LOOPBACK LEAD TESTS ALL THE MAJOR FUNCTIONS OF THE COMPUTER, EG SOUND, COLOUR, KEYS, RS232 MICRODRIVE ETC. AS USED BY REPAIR COMPANIES **£14.00**

KEYBOARD UPPER CASE INCLUDES, KEYS AND SPACEBAR £16.00
KEYBOARD LOWER CASE.....£5.00
KEYBOARD MEMBRANE.....£6.00
BUBBLE MAT.....£3.00
BACK PLATE £3.00
JS ROM SET £28.00
ZX8301 ULA £12.00
ZX8302 ULA £13.00
RAM 4164 150ns £5.00

SOFTWARE

TOUCH TYPIST	£12.00	MEDIA MANAGER	£40.00
TASK MASTER	£25.00	THE QUILL	£23.00
SPELLBOUND	£29.95	FRONT PAGE	£23.00
SIGN DESIGNER	£16.00	SCRABBLE	£15.00
CARTRIDGE DOCTOR	£13.00	ARCHIVIST	£30.00
EYEQ	£29.00	GST MACRO ASS	£43.00
ARCHIVER	£17.00	FINANCE MANAGER	£19.00
TURBO	£99.00	KEYDEFINE	£10.00
THE EDITOR (DP)	£30.00	TALENT WORKBENCH	£25.00
PROJECT PLANNER	£30.00	SUPERCHARGE	£49.00
QLIBERATOR	£70.00	PSION CHESS	£17.00
FORTH	£34.00	BETTER BASIC (DP)	£20.00
DECISION MAKER	£30.00	GST ASSEMBLER	£26.00
GARDNER	£20.00	HOME FINANCE	£20.00
TOOLKIT 2 (EPROM)	£34.00	REVERS!	£10.00
TECHNIQL	£49.00	OXFORD TRIVIA	£18.00

39 WRAY CRESCENT, ULNES WALTON, LEYLAND, LANCASHIRE PR5 3NA.
TEL: 0772 454328 PRESTEL MAILBOX 772454328
ACCESS AND VISA CARDS WELCOME.
PLEASE NOTE: ALL PRICES INCLUDE VAT & P/P.

David Batty

TASK MASTER

THE "ROLLS ROYCE" OF MULTI TASKING

NOT just another multi tasking program, but a complete front end memory management system for the expanded Sinclair QL.
ON SCREEN CALCULATOR with many features includiung delta percentage mark up/down.
POWERFUL FILE MAINTENANCE of discs/microdrives.
NOTEPAD with word wrap and "send note" feature.
SOFTWARE RESET scans for, and warns of any vulnerable open files at close-down.
SERIAL PRINTER BUFFER, user configurable up to 32K.
DUPLICATE CODE SHARING allows you to load multiple copies of the same program without losing memory to additional code space. EG. 12 Abacus with 230K free.
COMMAND FILE MODULE easily creates files which will operate the machine in your absence. EG. Could load Quill,. fill in your address, type a standard letter AND print it. You will however have to post it yourself.
SOME are under the impression that Taskmaster consumes large tracts of memory.
FACT if you only wish to use the multi tasking module then that is the only one that you need to load, thus using only a small slice of memory.
OBVIOUSLY if you wish to use any or all of the other integrated modules then progressively more memory will be used. THE CHOICE IS ALWAYS YOURS. Although complex in its actions Taskmaster is so user friendly that it was described as "VIRTUALLY IDIOT PROOF" by a major software reviewer.
AVAILABLE ON DISC OR MICRODRIVE AND STILL ONLY £25.00.

QL EDUCATION

QL EDUCATION

MONTH TWO: Dealing With Numbers

Number crunching

Leslie Fahidy looks at the teaching of mathematics in the second part of his new series on education.

We discussed last month a list of topics in which computers can be used for teaching. The list was by no means exhaustive but it showed how many useful tasks we can tackle with the QL. I said we would start a detailed discussion of those topic areas, mentioning the many possibilities of each field as well as the traps for which we must keep our eyes open.

Now I look at the teaching of mathematics and offer advice on the subject. Some years ago I was running a series of lectures for school teachers in a London Education Authority area on educational programs of the then newly-acquired school computers. A member of the audience remarked that computers must be suitable for teaching mathematics, since they can manipulate numbers so easily.

The conclusion was correct but the reasoning was not. Not only can computers manipulate numbers easily; that is the only thing they can do but their ability to manipulate numbers does not make them suitable for teaching mathematics. It is the display facilities they possess and their ability to repeat actions which makes them a good teaching tool.

First factor

If we want to write educational programs in the field of mathematics the first factor we must consider is the level at which the program should operate. In theory, there is no good reason why an educational program could not be used to teach the simplest rules of adding numbers, or the most complex mathematics you may require when doing post-doctorate research. In practice, however, we had better restrict the mathematics we teach on a computer to the lower levels of the subject.

To start, as the level increases so the number of people who can write such programs decreases. That also applies to the number of people who would want to use them — an important point since, I assume, we are talking of writing commercially-viable software. There seems little point in writing a program to teach your son how to solve quadratic equations; in the time it takes you to write the program you could have taught him the subject.

A second consideration is that if you found mathematics difficult you are unlikely to be studying it at higher levels; if it is easy and natural to you, you will probably prefer to learn it from books. I am not suggesting that the computer should not be used as a tool in mathematics — only that, beyond a certain level, it may not be the best vehicle for teaching it.

Instead of trying to rise to the heights of teaching degree-level mathematics on the QL, for commercial reasons alone I would restrict my activities to the lower levels of the subject. There is no shortage of topic areas there and, I believe, there would be no shortage of people interested in such programs.

My advice to the would-be educational software writer in the field of mathematics is not to go above O level or its equivalent GCSE standard and concentrate on the mathematical requirements of the eight to 14 age group. While talking of level, I feel I must mention a special point. In my schoolboy days it was the accepted custom that parents helped their children with their school work, to a point anyway. The trouble today is that too many parents have not even heard of many of the topics which their children are expected to learn at school, so they are unable to help.

Needless to say, I am referring to topic areas in modern mathematics where many of the fields are unknown even to those parents who had studied mathematics to A level and beyond in their day. Modern mathematics is a particularly promising field for the person who wants to write educational software in this area.

Leaving the level aside, what are the topic areas in which one should make an effort in writing good educational programs? I split the areas in three groups, based on level, and they are:

At the lowest level, age six to 12. The four rules, using whole numbers, have been published in two programs in *Sinclair QL World* in the past on multiplication tables and they may serve as a guide to the method of treatment which I would recommend.

Suitable age

In those areas the accent is on presentation; it must suit the age range of the learner and you must make every effort to make it interesting and enjoyable. Use a large display, plenty of colours, employ flashing characters here and there and, most important, praise the learner for every success.

The learning of the four basic arithmetic rules and the tables is essentially rote-learning and as such it certainly is not interesting or exciting. When writing such programs, the programmer must make every effort to please the learner as much as possible.

Telling the time. You can see an example of a clock program in Microdrive Exchange and if you followed the first educational series you will

QL EDUCATION

MONTH TWO: Dealing With Numbers

remember that I discussed it in detail there. The fact that a program, together with many others, already exists for the QL should not discourage you from writing your own. There are very few topics of which any of us is likely to think in the field of educational computing which have not already been tackled by someone else.

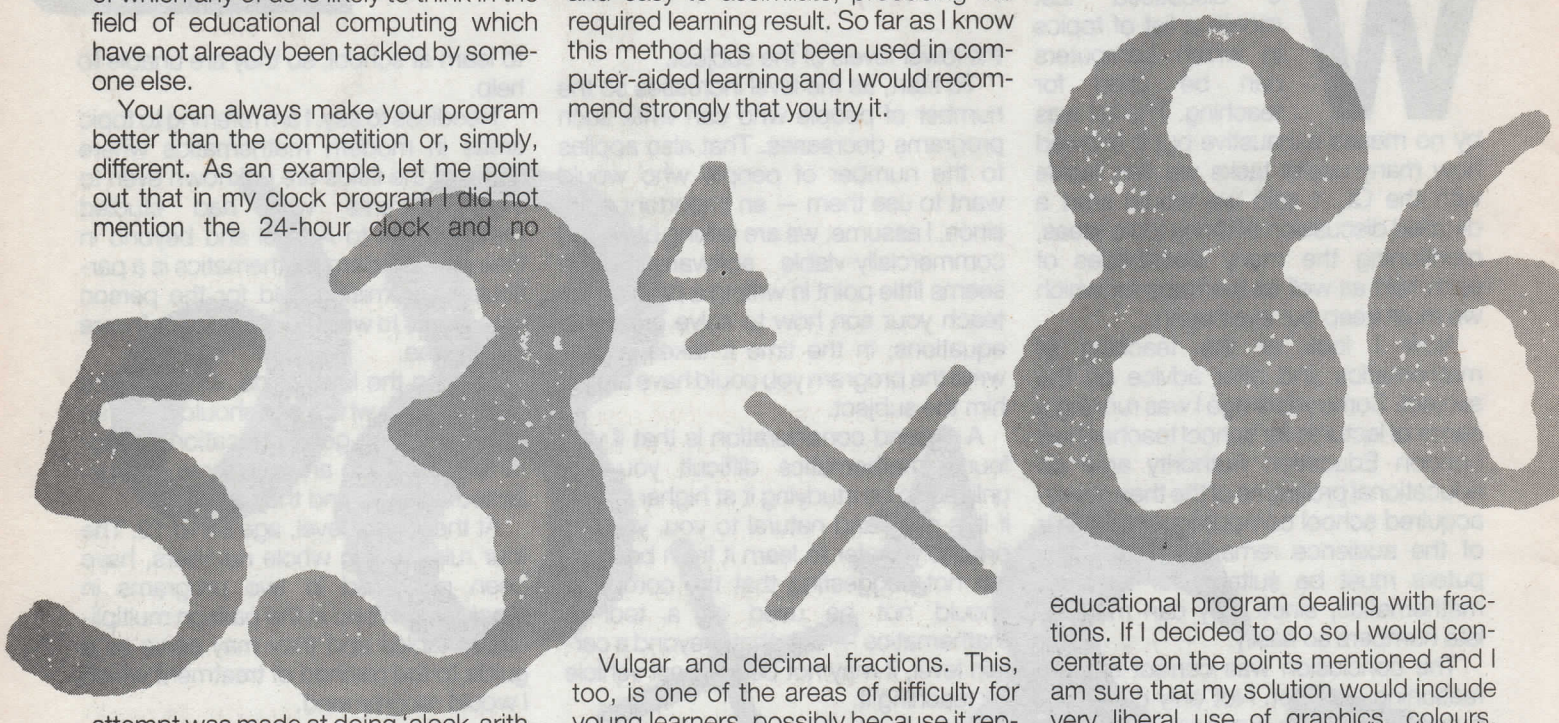
You can always make your program better than the competition or, simply, different. As an example, let me point out that in my clock program I did not mention the 24-hour clock and no attempt was made at doing 'clock-arithmetic'; either one of those two ideas could materialise as a very successful clock program.

Percentage calculations. For some reason or another, most young pupils find percentage calculations difficult. I suppose it is not really the calculation alone which is difficult — rather the fact that they have to learn a new concept. I have often taught percentages from scratch and found it helped to give an introduction to show why percentage calculations are needed.

Generally, if you convince the learner that what you are about to teach is important, not merely an exercise to make the lives of students more difficult, they are more willing to learn it. When teaching percentages I usually start with examples on comparing related quantities. Something along the lines of Joan has 12 questions correct, while 14 of Jim's answers were correct. Who showed a better performance? Obviously, we cannot answer that without knowing how many questions each child had to answer.

That leads us to the idea of comparing on the basis of what fraction of the whole it is. Up to that point I have not mentioned the word percentage. At this stage, deliberately, I introduce a further problem. If Joan had 12 correct out of 15 while Jim had the correct answer to 14 out of 20 questions, Joan did better in the test. We work out a few examples along those lines and when everybody is disenchanted with evaluating and comparing fractions whose denominators are different, I suggest that we should, in each case, state how many would have been correct if there were a fixed number, say 100, questions.

Children find this approach logical and easy to assimilate, producing the required learning result. So far as I know this method has not been used in computer-aided learning and I would recommend strongly that you try it.



Vulgar and decimal fractions. This, too, is one of the areas of difficulty for young learners, possibly because it represents a quantum leap compared to learning the four rules. While you are dealing with the basic rules of arithmetic you are on firm ground; it is merely a question of knowing the rules and applying them.

Even multiplication tables, considered to be the most difficult of the basic elements of arithmetic, are comparatively easy, even though they represent a great deal of learning. When you start dealing with fractions you must make a number of decisions and act accordingly. It is decision-making which seems to be the source of difficulty.

You must decide whether a fraction is decimal or vulgar and apply different treatment to it accordingly. If it is a decimal fraction, you have to remember a set of fairly complex rules and apply them correctly. If it is a vulgar fraction, apart from having to choose from a complex conglomerate of rules you must decide whether or not their denominators are different. If they are, you must decide whether or not that calls for a different treatment. In multiplication and division it makes no difference; in addition and subtraction it matters.

Younger generation

I am trying to put the case of the younger generation, showing what it is that they find difficult and why. If you want to write successful educational software you must concentrate on the question of what causes the difficulty and how you may overcome it. We all know that fractions are easy; the difficulty is that children do not seem to take our word for it.

As yet, I have not written a serious educational program dealing with fractions. If I decided to do so I would concentrate on the points mentioned and I am sure that my solution would include very liberal use of graphics, colours, even flashing colours. I would use every means at my disposal, including sound, even though I am as non-musical as it is possible to be, to make the topic more interesting.

Money. Apart from its general usefulness in life, dealing with money is another of those topics which every child should master at school. You have probably noticed already that mathematical education at the lower end of the age group divides naturally into two distinct areas.

There are certain topics which every child should learn at school, because those topics are necessary in everyday life. On the other hand there are topics which are intended for those who might become mathematicians, scientists, engineers or professionals in some numerate field.

All the topics considered so far, including the topic of money, definitely belong to the first group. You could go through life easily even if you could not solve a second-order differential equation but you must be able to apply the four rules of arithmetic, use fractions and percentages and be able to tell the time, almost irrespective of what you intend to do later in life.

In this issue you will find a cartridge of mine in Microdrive Exchange which is one possible solution to the question of teaching children how to deal with money. It includes only the most elementary necessities but, I think, it does that job reasonably well. That does not mean that you could not improve it — I will even suggest ways of doing so.

Since the elements of dealing with money are taught at a young age, the program would certainly benefit from graphics. Include routines which draw the different coins, approximately true to shape, and I am sure that children will consider it as a game program rather than a piece of educational software. That, after all, is our aim.

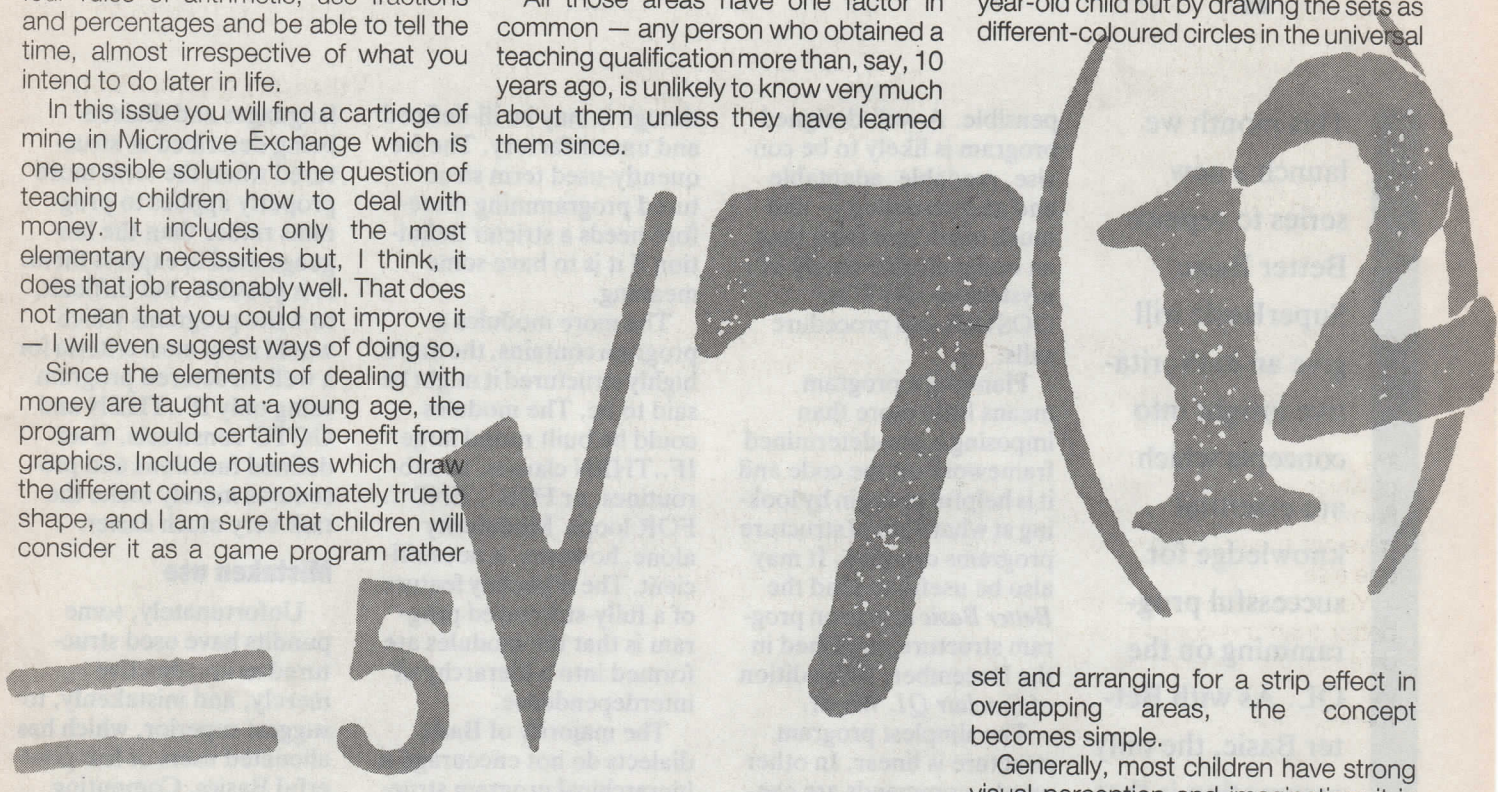
There are many other topics which are taught in mathematics in middle schools; they are important but the ones outlined are the absolute essentials every child should learn.

At a higher level, age 10 to 14, we are dealing with topic areas which are, or will be, needed by every child who intends to proceed to most forms of higher or further education. I should have said children whose parents intend them to do so, because most children intend to do very little at this age.

Note that I did not just say further or higher education in a numerate discipline but, intentionally, I included all higher and further education in any sphere. It is debatable whether or not this is as it should be but children do not stand much chance of academic or professional educational beyond the age of 16 unless they have a reasonable pass in O level mathematics — or its GCSE equivalent when that becomes a working reality.

Modern mathematics. I start deliberately with this topic since it is the field where the greatest need exists. Also it is the field with the greatest difficulties. If you can write good educational software in modern mathematics, on any computer including the QL, you should not be short of work. The areas of the greatest importance are set theory, number bases, matrices and determinants and the theory of functions, including inverse functions.

All those areas have one factor in common — any person who obtained a teaching qualification more than, say, 10 years ago, is unlikely to know very much about them unless they have learned them since.



Another source of difficulty when teaching these topic areas on the computer is that the concepts are entirely new to the children. Even if we assume that they can perform perfectly all the tasks mentioned previously, number bases, matrices, function theory, and so on seem, at first, divorced from anything which happened previously. They are not divorced from each other but for various technical reasons concerning how we teach mathematics they appear to be relying on different concepts.

Since, much more than in other fields, we are first and foremost teaching concepts, the method must reflect that requirement. Instead of praising the learner or sustaining interest, which is still important at this age, presentation of ideas becomes the most important factor.

Problems solved

Luckily we not only have problems but also solutions. Again, the graphics capabilities of the QL are of utmost importance and they should be used regularly. As an example, consider the task of writing a program to teach the multiplication of two 2 X 2 matrices. The rules of matrix multiplication are complex and boring but if we print a coloured arrow superimposed on the first matrix and show how it travels along the rows, while a second arrow, coloured differently, keeps moving along the column of the second matrix, the concept of matrix multiplication can be made to appear to be very simple, which it is.

Alternatively, consider set theory. The terms 'AND', 'OR', 'NOT', 'XOR' are fairly complex to the minds of a 13-14-year-old child but by drawing the sets as different-coloured circles in the universal set and arranging for a strip effect in overlapping areas, the concept becomes simple.

Generally, most children have strong visual perception and imagination; it is the abstract logic of it which presents the difficulties. We must utilise this fact and present as many visual images as we can possibly use.

Because, particularly in this last section on modern mathematics, I have given a number of mini algorithms I am beginning to run short of space. There are more points I wanted to mention in connection with teaching modern mathematics on the computer, as well as a number of other topics besides modern mathematics in this age and ability range which should be discussed. Luckily, there is always a next month and in the October issue I will continue the discussion. When we have finished with mathematics, we shall deal with other subjects.

I hope that you will also be doing some thinking in the meantime. First and foremost, I hope that you will try some of the ideas in this instalment. You will probably have noticed from the style of this article that I am not dealing with the elementary aspects of the task, i.e., how to write code. That is what I promised you at the beginning but do not overlook the importance of code writing.

If you feel you need to revise SuperBasic, now is the time to do so, before we go more deeply into the algorithm aspect of educational software. I hope that you will also think along the lines of which topic areas you would like to see covered in the series.

SuperBASIC

This month we launch a new series to replace Better Basic. SuperBasic will give an authoritative insight into concepts which are essential knowledge for successful programming on the QL. As with Better Basic, the only assumption is that readers are familiar with the User Guide. Mike Lloyd begins with a two-part appraisal of program planning.

Just as a builder can build a ramshackle house, a programmer can write a ramshackle program. The difference between a shack and a mansion might not lie in the quality of building work but in the architect's design. Programmers are both architects and builders but the former skill is all too often forgotten in the rush to practise the latter.

For anything more than the most trivial SuperBasic program, some pre-planning, a systematic approach to code writing and a clear program structure are indispensible. A well-designed program is likely to be concise, readable, adaptable and easy to debug — and much more satisfying than an impenetrable tangle of mysterious GOTOs, GOSUBs and procedure calls.

Planning a program means little more than imposing a pre-determined framework on the code and it is helpful to begin by looking at what kind of structure programs can take. It may also be useful to read the *Better Basic* article on program structure published in the December, 1986 edition of *Sinclair QL World*.

The simplest program structure is linear. In other words, commands are executed in strict line number order. Basic interpreters assume a linear program structure unless told otherwise. Although simple, a linear structure is potentially very inefficient, a trivial routine to print 10 digits, for example, would require 10 commands.

All Basics therefore contain control constructs which divert the interpreter from its linear progress. The constructs are associated with a small number of keywords which can be grouped into three main classes — see inset; repetition, i.e., executing commands in loops; conditional clauses, i.e., executing some commands only if a certain condition is true; and digression, i.e., jumping to subroutines. Control constructs and the program lines associated with them form control structures.

Programs tend to consist of several control structures, perhaps interspersed with a few lines executed linearly, therefore every program can be said to have some kind of structure even though it may be ill-defined and unsatisfactory. The frequently-used term structured programming therefore needs a stricter definition if it is to have some meaning.

The more modules a program contains, the more highly structured it might be said to be. The modules could be built round large IF..THEN clauses, or subroutines, or FOR...END... FOR loops. Modularity alone, however, is not sufficient. The other key feature of a fully-structured program is that the modules are formed into a hierarchy of interdependence.

The majority of Basic dialects do not encourage a hierarchical program structure but some other languages compel programmers to write in this way. In C, for example, programs consist entirely of function definitions. One of the definitions must be called main to identify it as the first to be accessed when the program is run. Other program segments are called from within the main procedure and therefore are subordinate to it; they may, in turn, call other segments, and so on.

Direct command

The mandatory organisation of a C program can easily be mimicked using SuperBasic procedure and function definitions, provided main is entered as a direct command instead of RUN. Unlike C, SuperBasic also allows programmers to write unstructured programs if they wish, usually to their cost.

It would be too strict a requirement to say that user-defined procedures and functions are essential features of any structured program. It also leads to languages and dialects being described as structured when the term more properly applies to programs rather than the language used to express them. It is possible, but difficult, to write programs which would meet most criteria for a well-structured program using only IF..THEN and GOTO constructs. User-defined functions and procedures merely make the task very much easier.

Mistaken use

Unfortunately, some pundits have used structured as an adjective merely, and mistakenly, to suggest superior, which has alienated users of less powerful Basics. Computing philistines have taken the opposite view, that so long as the user's requirements are met and a program is bug-free its structure is of little relevance. Both groups perhaps fail to recognise that structure is just a means to an end and that while the success of a program depends on its performance, its performance depends in turn on its structure.

A program is like a Wild West movie set — what the film-goer sees is Dodge City while what really exists are plywood facades supported by props and struts. The scaffolding could be highly disorganised and so long as it did not collapse completely the viewer would be no wiser.

What a software user sees is a facade supported by the scaffolding of a program listing. A strong and coherent program structure might be of secondary importance and invisible to the user but without it the facade of the program might collapse.

It is surprising that many

programmers fail to make full use of the structured programming techniques possible with SuperBasic, seemingly determined to struggle with untidy and convoluted listings rather than learn structuring techniques. Some are reluctant to accept the necessary discipline of pre-planning when writing programs as a hobby. Others stubbornly retain habits learned on less powerful Basics, perhaps suspecting that structured programming is part of the jargon seemingly designed as much to exclude the unintiated as to describe useful computing concepts.

A structured program can maximise execution speed by minimising the number of re-directions. By avoiding duplication of code it is likely to be concise, decreasing loading time and increasing memory space for data. Errors should be less likely to occur and easier to trace and correct. The program is likely to be easy to modify and extend and the listing should be straightforward to read and understand.

Minimum effort

Because of those benefits, structured programming creates the conditions in which larger, more complex and more powerful programs can be written with the minimum of extra effort. In short, structured programming is programmer-friendly.

Structuring, then, is simply a means to an end, a way of producing successful programs with the minmum of effort. Structured programs have two essential features; they are composed largely, or preferably entirely, of user-defined program segments and those segments are arranged in a hierarchy.

So how is a structured program best written? What might be described as linear programming, starting at the beginning and coding until the end is reached, is unlikely to be effective unless the whole problem and its solution are clear in the programmer's mind. Such clarity of vision is unlikely for anything but the simplest application and therefore the problem must be broken into mind-sized pieces which can be tackled separately. That is achieved most easily by taking advantage of a program's modular and hierarchical nature. One way of doing this is called top-down planning.

First car

To take an example far removed from computing, imagine how Karl Benz might have designed the first motor car. He might first have defined his objective as "the construction of a buggy powered by a mechanical engine as powerful as a horse". After carrying-out feasibility studies the project was probably divided into a number of distinct areas such as engine, suspension, drive train and vehicle body, themselves broken down into sub-components. Satisfied about the general structure of the car, he could then afford to concentrate on the technicalities of each component in turn.

Planning in that way is a recursive technique. Starting with the broad outline of an ideas, it is broken into a number of components. They are divided and subdivided repeatedly until each tiny part of the project has been identified and dealt with. Figure one shows a program planning guide written as a mock SuperBasic procedure.

Although all programs are different, the way in which they are planned changes little. The first step is to translate the idea, such as a home finance program, into something more tangible by defining the problem more precisely, for example a program which accepts, stores and analyses information relating to a bank current account. The definition helps to establish the boundaries of the project so that a routine to control standing orders would be included but an investment analysis module would not.

Some programs never seem to be finished because there is always some refinement or extra features which can be included. Professional programmers cannot allow that to happen because they must produce programs to order and on time. Therefore they develop a clear understanding of the aim and ruthlessly exclude routines which do not go towards meeting it.

Main concept

Once the main concept of the program has been decided, its feasibility needs to be considered. A variety of factors might make an idea unworkable and they are best identified before lengthy programming work begins.

The program input could be difficult to obtain. The effort of loading a home finance program to record each cheque transaction, for example, tends to outweigh the value of the program output. Alternatively, the input might be too massive to be worth typing-in, or too great to be held in the computer memory, or impossible to obtain.

Even if the input were readily available and could be stored adequately, the problems associated with manipulating the data might still be intractible. Someone considering writing a chess-playing program, for example, would need to be well-versed in a variety of data storage, searching and evaluation techniques before program coding could begin.

Finally, the output might prove to be the stumbling-block. A smoothly-revolving, three-dimensional cube, for example, might be impossible to code in SuperBasic.

It is good practice to think

```
DEFine PROCedure PLAN (problem)
  IF problem = mind sized THEN
    SOLVE problem
  ELSE
    RESEARCH problem
    IF problem = NOT feasible THEN STOP
    DIVIDE problem INTO segments
    FOR x = 1 TO number of segments
      PLAN segment (x)
    END FOR x
  END IF
END DEFine PLAN
```

Figure 1

through all the steps a program will need to go through and note any likely trouble-spots. Even though much of the coding eventually will prove to be straightforward, every program will have its particular challenge which will need to be researched thoroughly before programming time is invested in the project.

Worth continuing

Experienced programmers can assess quickly whether a project is worth continuing and thus avoid spending fruitless hours reaching for an impossible goal. Programmers without experience tend to gain it only by rushing on enthusiastically where the wiser ones decline to go.

When feasibility is in doubt, some alternative strategies are to consider using another language — or a compiler; to research new programming techniques; to re-assess the main concept of the program; to shelve the project until one is more experienced; or, if all else fails, to give up entirely.

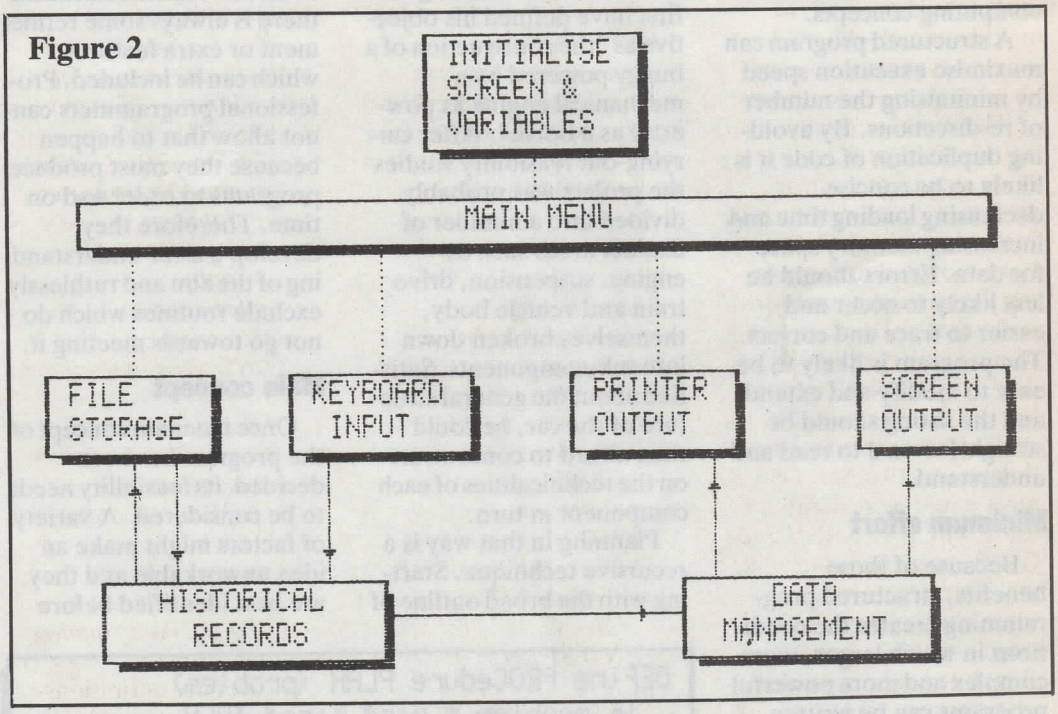
Once an idea is deemed to be feasible the next step is to analyse the essentials of the program. Many of them will have surfaced during the feasibility study. Such factors, however, as the limitations which one part of a program might impose on another are now important and they might not be obvious without some form of diagrammatic representation.

The first stage in analysis is to identify the major components of the program. For a typical personal finance program they might be:

INITIALISE SCREEN AND VARIABLES
MAIN MENU
FILE STORAGE
KEYBOARD INPUT
PRINTER OUTPUT
SCREEN OUTPUT
DATA MANAGEMENT
HISTORICAL RECORDS OF ACCOUNTS

These will be referred to as program segments. More information is conveyed if the segments are represented as blocks on a page and their interdependence shown by lines between them, as demonstrated in figure two, which is a cosmbination of a menu structure diagram and a simple dataflow diagram.

Figure 2



For the moment, each segment can be regarded as a black box in which processes occur mysteriously. Only the input to the boxes and the output from them is important. The dataflow diagram can be made more detailed by noting the exact nature of the data moving between the segments.

If the data held in historical records was in a compressed format, the keyboard input segment would need to include a suitable encoding routine. The diagram indicates that the data could be filed in its compressed state. Because ASCII format is required by both printer and screen output the logical place for a decoding routine would be in the data management segment. Different relationships between these segments might lead to different assumptions about the data.

Clear idea

By now a clear idea of the shape of the program as a whole should be emerging and it should therefore be safe to take one's mind off the entirety of the project and begin to concentrate on increasingly smaller aspects until a level is reached at which coding SuperBasic statements can begin.

The first step is to choose one of the segments, consider it as if it were an independent program and repeat as many of the foregoing stages as are appropriate. Whereas a program is divided into segments, each segment is divided into modules, each of which might eventually become a procedure or function definition. Throughout this article, a procedure or function definition is referred to either as a module or as a routine.

Like the segments of a program, modules will form some kind of hierarchy and they can be described according to their relationship with other modules. A slave routine is one which is referred to by only one other module. Modules referred to by many other modules are described as utility routines. The distinction is important, because a slave module might share variables with its master module while variables in a utility routine might need to be declared as local or passed as formal parameters.

Other modules can be referred to collectively as major routines but it might be helpful to distinguish control modules, mainstream modules and error-handling routines. Of course, some of the character of routines will be determined by the kind of segment to which they belong; a screen-handling segment might contain graphics routines and an input segment contain keyboard reading routines, and so on.

The planning steps to this point would be the same, no matter in what language the program is to be written. It should be possible to take a program plan and implement it in any high- or low-level language.,

Next month's article deals with planning and implementing individual procedure and function definitions. Obviously, that stage becomes more language-specific.

Super BASIC

Control structures

The control structures implemented in a computer language, often using only a handful of keywords, make a disproportionately large contribution to the overall character and power of the language. They form a distinct sub-group separate from other keywords. Qdos, for example; does not include them in the name table in which all other keywords and user-defined names are stored.

There are generally agreed to be at least three main classes of control structure which can each be further sub-divided. Basic is not renowned for supporting a wide range of control structures and among personal computer dialects only BBC Basic and SuperBasic can claim to be highly-structured. It is interesting that almost all control structures could be re-written using only IF...THEN and GOTO constructs but while this is straightforward, when considering a simple loop it becomes more complex, but not impossible, when considering techniques such as recursion.

Class one: Repetition
CONSTRUCTS:

REPeat ‹name›
‹statements›
**EXIT ‹name›*
**NEXT ‹name›*
END REPeat ‹name›
FOR ‹variable› = ‹value› TO ‹svalue›
**STEP ‹value›*
‹statements›
**NEXT ‹variable›*
**‹epilogue statements›*
END FOR ‹variable›

Repetition includes iteration and recursion. FOR...NEXT loops are, strictly speaking, iterative because the test for leaving the loop is defined at the beginning of the construct. REPEAT...UNTIL loops — not implemented in SuperBasic — are deemed repetitive because the test for leaving is located at the end of the loop. A more valuable distinction between them is that a FOR...NEXT loop is explicitly finite, whereas a REPeat loop is implicity infinite.

Higher form

Recursion is a higher form of repetition because it requires procedure definitions. It can be simulated in ordinary Basic dialects using stacks to hold the equivalent of local variables.

Class two: Conditional clauses
CONSTRUCTS:

IF ‹condition› THEN
‹statements›
**ELSE*
**‹statements›*
END IF
*SELect ON ‹variable› = ‹value› *TO ‹value›*
‹statements›
END SELect

ON ‹variable› GOTO ‹line no›, ‹line no›,...
ON ‹variable› GOSUB ‹lines no›, ‹line no›,...

ASSOCIATED KEYWORDS:

GOTO
GOSUB
RETurn

The object of conditional clauses is to re-direct program flow to include or exclude statements according to the evaluation of a condition. The familiar IF...THEN construct allows binary decisions to be taken according to whether an expression is true or false, whereas the more powerful SELect construct permits an almost unlimited number of options based on the inclusion of a variable in a range of values.

To some extent, this is also true of the ON...GOTO/GOSUB constructs. In common with Spectrum Basic, computed GOTO and GOSUB line references are allowed as an additional method of making decisions.

Class three: User-defined procedures and functions
CONSTRUCTS:

DEFine PROCedure ‹name› (‹parameter›, ‹parameter›..)*
**LOCal ‹variable›, ‹variable›...*
‹statements›
**RETurn*
*END DEFine *‹name›*

*DEFine FuNction ‹name› *(‹parameter›, ‹parameter›..)*
**LOCal ‹variable›, ‹variable›...*
‹statements›
RETurn ‹value›
*END DEFine *‹name›*

Procedure definitions would be no more than labelled subroutines if it were not for the ability of SuperBasic to coerce values, pass parameters and declare local variables. Coercion stems from he need for formal parameters to adopt the variables. Coercion stems from the need for formal parameters to adopt the variable type of the actual parameter which they represent. Recursion is the ability of a definition to call itself. Local variables exist only within the definition in which they are declared.

User-defined procedures and functions can be considered as independent programs, as SuperBasic extensions or as superior subroutines. They are perhaps best seen as the building blocks from which programs are constructed.

Note: Asterisks by control structure commands indicate optional statements.

Colin Opie looks at some basic digital and analogue circuits in the final part of this series.

In this concluding part of the Connexions series we look at some basic digital and analogue circuits and how the outputs from those circuits can be collected and monitored by the QL using the Qontrol-II board. The purpose is not to teach you all you need to know about interfacing projects. The major reason for this approach is that any other objective would be impossible to achieve.

If you know nothing about interfacing and/or electronics, but are interested, I would suggest you read this article and obtain a few books and a knowledgeable friend. Better still, why not attend an evening class at your local college? On the other hand, if you have had experience with projects on other computers, you should have no difficulty using the information to start using your Qontrol-II board to the full.

The A/D device on the board is extremely easy to use and involves reading one of eight memory locations. The versatile interface adapter Qontrol-II uses is a device called a 6522. Some will know of this device and the many uses to which it can be put. Essentially, it is a programmable device which has 16 internal control and data registers.

If you want to program this chip in any special way you will have to obtain a data sheet or a book which details its operation. Remember, though, that the Qontrol-II ROM contains some very useful routines. The routines would be capable of supporting non-complex control requirements.

The A/D chip uses eight addresses, one for each A/D channel available. Each channel can sample data at more than 1,500 samples per second, automatically and continuously. The eight channels are sampled in strict rotation and therefore if you tied a signal to all eight input lines you could sample a signal at precisely 12,500 samples per second.

One way of reading the A/D channels is to use the function AD_BASE which will return the base address of the A/D chip. You could then PEEK the address for channel 0 and the next seven addresses for channels 1 to 7 respectively:

```
100 REMark Display A/D input values
105 ad=AD_BASE
110 FOR chan=0 TO 7
120   PRINT PEEK(ad+chan)
130 NEXT chan
```

Alternatively, you can open a channel named 'atod0' to 'atod7'. Whenever you want to read a value from that channel you would do so in the normal way for SuperBasic:

```
100 REMark Monitor A/D channel 5
110 OPEN#3,atod5
120 REPeat loop
130   PRINT CODE(INKEY$(#3,-1))
140 END REPeat loop
```

Which number you choose would probably depend on the number of channels being used and programming style. The first method is not so elegant but saves a good deal of channel definition memory space and so is useful if all eight channels are to be monitored. The second method is more elegant but would probably be used only when a few channels are being used.

Monitoring the channels in graphical form can, of course, be programmed in any way you wish but the ROM contains a procedure called DIGISCOPE which displays up to four A/D channels as specified — see figure one. This is useful if you need to get a view of the trend of the input rather than obtain calibrated values from some piece of equipment. The routine is called using the form:

DIGISCOPE(waveforms,channels,rate)

The variable 'waveforms' contains the number (1 to 4) of inputs you wish to monitor. Variable 'channels' is a coded number from 1 to 255 specifying at which channel(s) you want to look. Each channel is allocated a flag bit in the 'channels' byte. Channel 0 is flagged by bit 0, channel 1 by bit 1 and so on. The 'rate' variable is the number of 1/50 second — or 1/60 on some foreign QL's — you want to delay between samples. As an example suppose we wanted to monitor channels 6, 3 and 1 once every second. Our 'channels' byte would be given by:

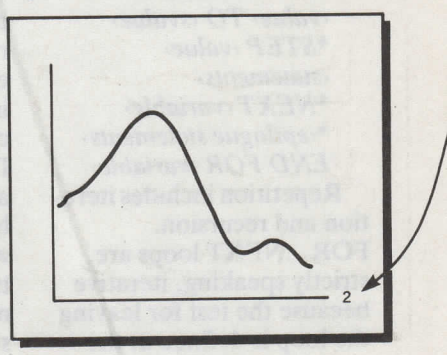
```
BIT/channel 7 6 5 4 3 2 1 0
set/reset   0 1 0 0 1 0 1 0 = 74
```

and so our call would be DIGISCOPE(3,74,50). That is perhaps a little unreadable but a well-written program could make life more pleasant:

```
100 REMark Display A/D channels graphically
100 sec=50
120 ad0=1:ad1=2:ad2=4:ad3=8...ad7=128
130 DIGISCOPE(3,ad1+ad3+ad6,1*sec)
```

The display is produced using 400

Figure 1. Digiscope displays.



WITH ONE CHANNEL

points along the horizontal axis. With a one-second delay the display would therefore take six minutes and 40 seconds to complete. The completion time is independent of the number of channels being displayed. At the end of a scan the display will halt and wait for one of three key entries:

ENTER — Clear screen and scan once more

SPACE — Continuously scan until a key is pressed

ESC — Exit from DIGISCOPE

Sometimes any of those three approaches could prove to be too slow for what we need. That is partly because of design, in the case of DIGISCOPE, and partly because of the speed of SuperBasic when reading directly. One option is to revert to assembly language but it is not an option available to everyone.

A fourth A/D monitoring capability is therefore available, ADGRAB. It will collect a specified number of samples from a specified A/D channel, placing the readings into a buffer. The sample rate will be 1,000 times a second, i.e., every millisecond. The call is of the form:

ADGRAB(chan,samples,buffer)

where 'chan' is the channel number, 'samples' is the number of samples to collect and 'buffer' is the address of the storage area. It is the programmer's responsibility to ensure that the buffer area is big enough:

```
100 REMark GRAB 400 samples from A/D ‡2
105 REMark and display the waveform.
110 buffer=RESPR(400)
120 ADGRAB(2,400,buffer)
130 OPEN‡3,SCR_400x100a40x40
140 FOR x=0 TO 399
150     POINT‡3,x,INT(PEEK(buffer+x)/2.55)
160 NEXT x
170 CLOSE‡3
```

Note that in this example the total capture time would be just under half a second. The fastest time for DIGISCOPE would be eight seconds.

What about the hardware side of the coin? The A/D socket has a 10-volt reference connection which corresponds to a full-scale reading. In other words, if 0 volts are at the input of the A/D we will get a reading of zero. If 10 volts are at the input we will get a reading of 255 — the A/D chip works with 8-bit resolution. So long as we vary the input to the A/D channels between 0 and 10 volts we will obtain a full range of conversion values.

Figure two shows an extremely simple light meter circuit which could be attached to one of the Qontrol-II analogue inputs. When in the dark the impedance — resistance — of the ORP12 LDR is several megohms, which means that most of the applied reference voltage will appear across the LDR. The input voltage to the A/D chip therefore will be small and a low digital reading will be obtained.

When placed in strong light the resistance of the LDR drops to a few hundred ohms and most of the reference voltage will therefore appear across the constant 4k7 resistor. The nett result will be that we obtain a high reading from the A/D channel. Build the circuit and use

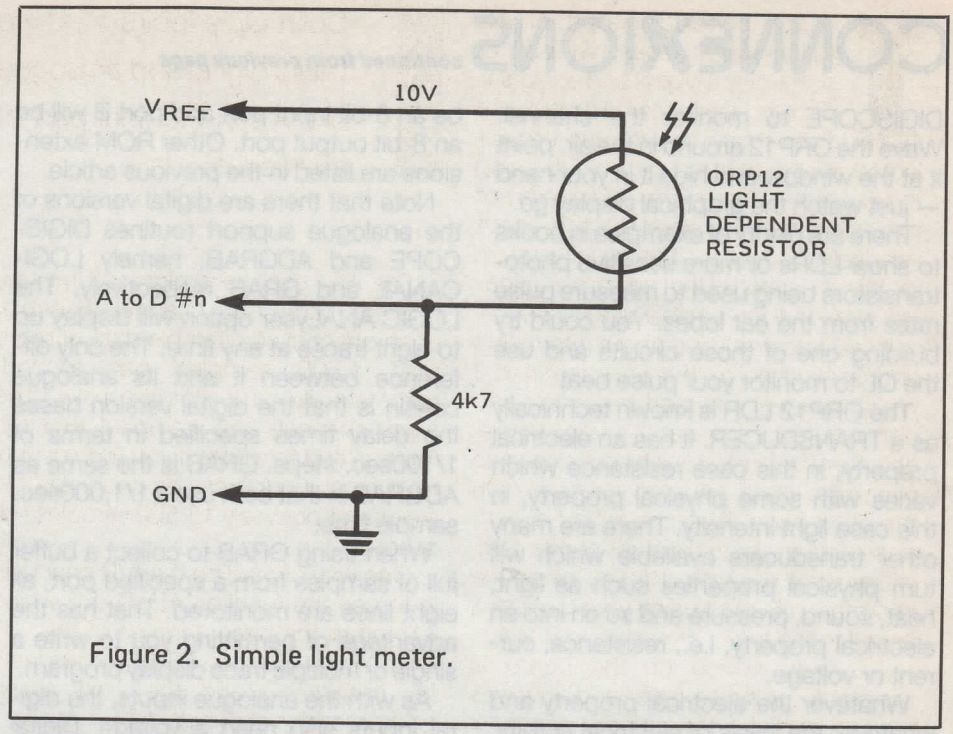
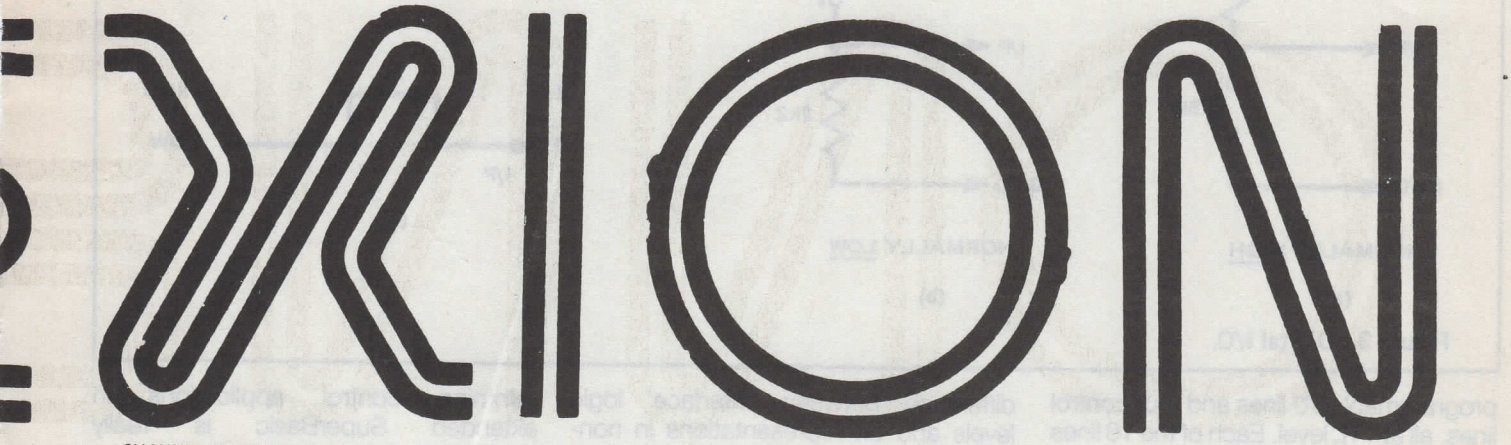
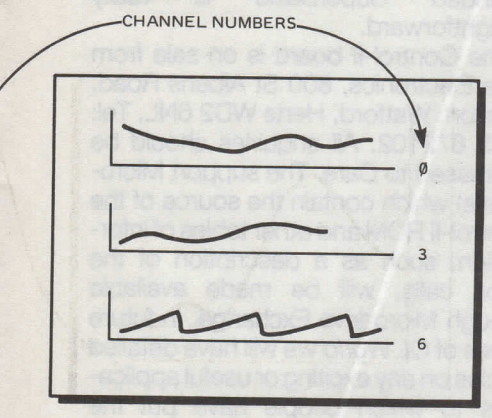


Figure 2. Simple light meter.





WITH THREE CHANNELS

CONTINUED OVER PAGE

continued from previous page

DIGISCOPE to monitor the channel. Wave the ORP12 around in the air, point it at the window and hide it in your hand — just watch the graphical display go.

There are plenty of examples in books to show LDRs or more sensitive photo-transistors being used to measure pulse rates from the ear lobes. You could try building one of those circuits and use the QL to monitor your pulse beat

The ORP12 LDR is known technically as a TRANSDUCER. It has an electrical property, in this case resistance which varies with some physical property, in this case light intensity. There are many other transducers available which will turn physical properties such as light, heat, sound, pressure and so on into an electrical property, i.e., resistance, current or voltage.

Whatever the electrical property and whatever the levels of electrical activity, we have to supply the Qontrol-II A/D inputs with a voltage range between 0 and 10 volts. The circuits required may be simple or complex, as dictated by the application.

Last month we learned what a TTL digital signal looked like and noted that there is very little power capability in these signals. The 6522 VIA has 16 programmable I/0 lines and four control lines, all at TTL level. Each of the 16 lines can be programmed as a TTL input or a TTL output independently. That gives us plenty of scope for matching project requirements with available resources. The chip is a complex one as well as being versatile and it would not be sensible to try and support all modes of operation in the Qontrol-II ROM. The software support for this device concurs with this.

The 16 digital lines are grouped into two 'ports' labelled 'A' and 'B', each having eight bits. There is a command 'PORTSET' which will set up the VIA to look like a simple I/0 device. Port A will be an 8-bit input port and port B will be an 8-bit output port. Other ROM extensions are listed in the previous article.

Note that there are digital versions of the analogue support routines DIGISCOPE and ADGRAB, namely LOGICANAL and GRAB respectively. The LOGIC ANALyser option will display up to eight traces at any time. The only difference between it and its analogue cousin is that the digital version bases the delay times specified in terms of 1/100sec. steps. GRAB is the same as ADGRAB in that both use a 1/1,000sec. sample time.

When using GRAB to collect a buffer full of samples from a specified port, all eight lines are monitored. That has the advantage of permitting you to write a single or multiple trace display program.

As with the analogue inputs, the digital inputs also need a voltage. Digital inputs are best kept to close to 0 volts (logic LOW) or close to 5 volts (logic HIGH) with not much between. When we monitor these lines or set their value if they are output lines from Qontrol-II, we are talking only about the state of a single bit.

It can either be at logic '1' (HIGH) or logic '0' (LOW). Conceptually there is no difference between 'interface' logic levels and bit representations in non-control programs. So if you are content with the latter you should be able to cope well with the former. In practice there are many times when a line is 'idle' at a particular level and signals that 'something of importance is happening' when the line goes to the other level.

Figures 3a and 3b show this kind of thing using switches as the example 'event creator'. It does not have to be switches. If the external device generates its own TTL voltage levels, e.g., an op-amp, there will also be no need for the +5V reference from the Qontrol-II board.

There is a correct and an incorrect way to set and re-set individual bits of a port so as not to upset the other bits. Fortunately you need have no concern about this because the Qontrol-II ROM contains the two routines TSTBIT and SETBIT:

level=TSTBIT(port,bit)
SETBIT(port,bit,level)

where 'port' is 0 for port A and 1 for port B, 'bit' can take any value between 0 and 7 to select the logic line required, and 'level' is the state of the line or bit. Suppose we wish to output a 'negative-going' pulse — see figure 3c — on line 3 of our output port (B). This can be achieved easily:

```
100 SETBIT(1,3,1): REMark Ensure line is high
150 SETBIT(1,3,0): REMark Send out a low pulse
160 SETBIT(1,3,1)
```

What could be simpler? Checking for a complete pulse coming in — see figure 3d — on line 4 of our input port (A) is just as easy:

```
180 IF TSTBIT(0,4)=0 : GOTO 180 : REMark wait for high
190 IF TSTBIT(0,4)=1 : GOTO 190 : REMark wait for low
```

Each application will have to be considered in the light of its difficulties and merits but I hope you can see that programming control applications in extended SuperBasic is really straightforward.

The Qontrol-II board is on sale from Care Electronics, 800 St Albans Road, Garston, Watford, Herts WD2 6NL. Tel: 0923 672102. All enquiries should be addressed to Care. The support Microdrive(s) which contain the source of the Qontrol-II ROM and other tables of information, such as a description of the Qdos calls, will be made available through Microdrive Exchange. In future issues of *QL World* we will have detailed articles on any exciting or useful applications to which people have put the board.

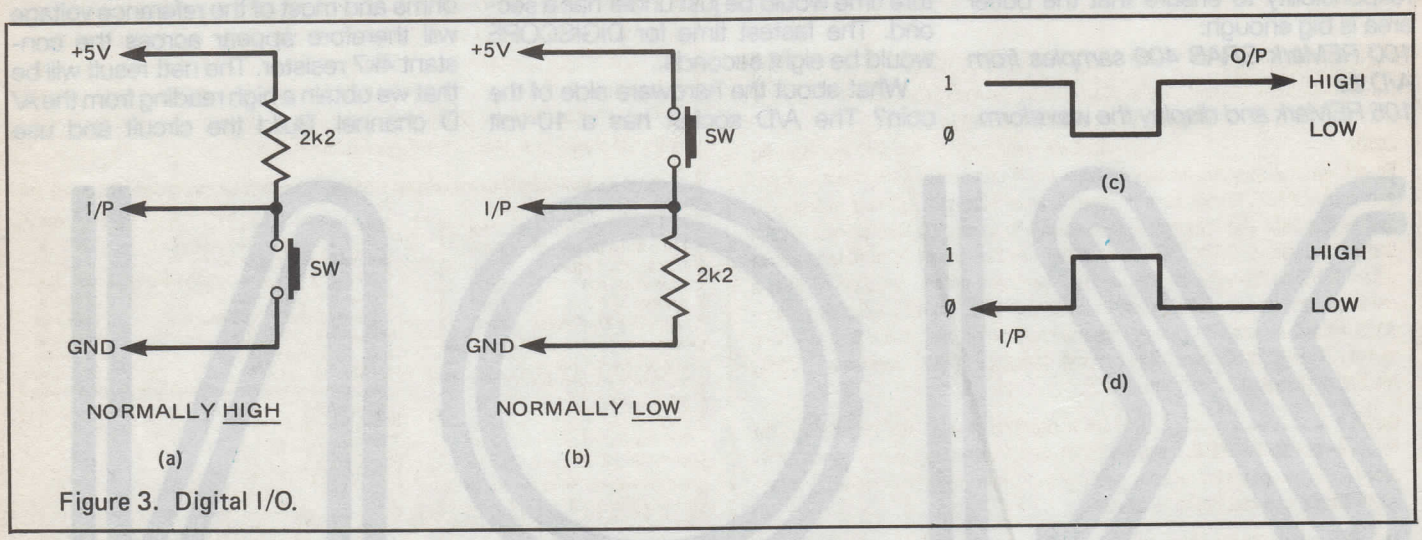


Figure 3. Digital I/O.

POWERFUL
YET EASY TO USE

Compware's own range of high quality software:

Task Swopper	£19.95
Copycat	£10.99
Book 128K	£5.99
1 to 1 Dump	£5.99

Microdrive Toolkits:

Basic	£9.99
Assembly Language	£24.95
Both on 1 cartridge	£24.95

Computer One's excellent range:

Pascal	£39.95
Forth	£39.95
Assembler	£29.95
Monitor	£19.95
Typing Tutor	£14.95

Miracle Hardware:

Trump Card	£195
512K Expandaram	£95

Prices include VAT and delivery.
Export orders welcome – write for terms.

See us on Stand 85/86 at the ZX MICROFAIR

COMPWARE

Ask for our full price list.

57 Repton Drive, Haslington, Crewe CW1 1SA
Tel: (0270) 582301 Tlx: 265871 (Quote: MAG70076)

WDSoftware

FOR THE SINCLAIR QL:-

JOSS **£15 on mdv or 5¼" flp, £17 on 3½" flp**

Forget syntax errors and mistyped names in file commands! Just move a cursor and press SPACE. Cursor keys or joystick allow access to up to 8 microdrives and all the discs your interface will handle, with up to 150 files on each. Scroll & print directories, COPY, DELETE or PRINT any file, select TV or Monitor mode before LOADing/RUNning a program. Use keyboard only to set date or label a medium. Easy to use with **Psion** or other software. No silly icons to learn – JOSS will tell you what it's doing! Mass copying/printing utilities and programmer's toolkit. Specify disc size, tracks and interface **(CST, Cumana, PCML, MicroPeripherals)** or microdrive only.

RefQL7 ..... £11 on 2 mdvs or 1 3½" flp, £9 on 5¼" flp

For use with **ARCHIVE 2,** contains 1300 useful references and a search/print program. Find programs, articles and reviews buried in the magazines on your shelf! Cheap updates of earlier editions.

Mdv Extension Cable (8" £5.50

Add ZX Microdrives to your QL.

Joystick Adaptor £4.99

FOR THE QL, SPECTRUM, (ALL), BBC & ELECTRON:-

WD Morse Tutor **£4 cassette, £6 mdv or 5¼", £8 3½"**

Teach yourself to read Morse Code. From absolute beginning to 18 wpm! Feedback on screen or printer. Random letters, numbers or mixed, 100 random sentences, many helpful features include phonetic speech via Currah Micro-Speech (Spectrum). Discs unsuitable for BBC B+.

We export hardware and third-party software. Ask for lists

PAYMENT:-

In advance, in **STERLING** on British bank branches, international Giro, Postal Order or **ACCESS/MasterCard.** Add £1 outside Europe for AIRMAIL.

FROM:-

WDSoftware (QLW),
Hilltop, St Mary, Jersey, C.I. Tel: (0534) 81392

TK COMPUTERWARE
your QL stockist
Tel: 0303-81-2801

UTILITIES

XREF	£15
Recover	£20
Cash Trader	£60
Finance Manager	£19
Decision Maker	£35
Entrepreneur	£35
Project Planner	£29
Home finance (Buzz)	£22
Investment Monitor	£20
Integrated Accounts	£100
QUIMI + mouse	£70
Librarian 128K	£20
Qspell	£25
Spellbound (Mdv or D)	£30
Archivist	£30
QUATS	£25
Copymate	£11
QMon	£20
Touch Typist	£12
QUIMP	£40
Key Define	£10
Super Sprite Generator	£29
Task Master Mdv or D	£25

QFLASH - All configurations Phone for latest price details

Front Page	£23
Front Page Extra	£38
Desktop Publisher	£70
Nucleon	£20
Turbo Quill	£10
Super Toolkit II (Eprom)	£30
Reconfigurable Toolkit II	£29
QRAM (Mdv or D)	£30
The Quill Adventure Writer	£22
Professional Astrologer	£70

LANGUAGES
Phone for details of our extensive range including Forth, Fortran, Lisp, Pascal and C.

TechniQL	£50
Technikit	£25
Eye Q	£29
GraphiQL+	£25
Viewpoint	£20
Concept 3D	£40
Pro CAD 3	£40
Sign Designer	£17
Better Basic	£24
Super Media Manager	£40
QLiberator (Mdv or D)	£60
Talent Assembler Workbench	£25
DevPac Development Kit	£30
Turbo (Mdv or D)	£99
The Editor	£30
The Editor Special Edition	£40
Computer One Monitor	£24

GAMES

The Heart of Gern £15.50 ● The Lost Pharaoh £15 ● Caverns £11.50 ● Jungle Eddi £15 ● Slime 3D £13 ● Mortville Manor £20 ● Fictionary £11.50 ● Hyperdrive £15 ● Wanderer 3D £20 ● Dragonhold £15 ● Classic Adventures £10 ● Jabber £8.75 ● Quboids £10 ● Dark Side of the Moon £16 ● Droidzone £10 ● Blocklands £10 ● Karate £15 ● Tankbusters £15 ● War in The East 39.95 ● Squadrons £16 ● Chess £17 ● Backgammon £13 ● Flight Simulator £19 ● Bridge Player 2 £20 ● Cosmos £14 ● Scrabble £15 ● Oxford Trivia £18 ● Snooker £13 ● Matchpoint £14 ● Type 22 £18 ● Area Radar Controller £13 ● Reversi £10 ● Strip Poker £15 ● Super Croupier £15 ● Super Reversi £10

PRINTERS

M/Peripherals MP165	£234
Canon PW 1080A	£279
Brother M1109	£230
Epson LX86	£278
Epson LX86 + T/feed	£300

DISC SYSTEMS

Dual 3.5 in. and Trump Card	£408.00
Single D/D 3.5in. drive	£125.00
Single D/D 3.5 in. & 0K SQB Interface	£225.00
Dual D/D 3.5 in drive	£209.00
Dual D/D 3.5 in. & interface	£295.00
Dual 3.5 in. & SQB	£408.00
Cumana disc interface	£ 86.25
768K Trump Card	£199.00

MODEMS

Miracle Modem	£49
Modaptor +	£39
Tandata QCOM	£94
Astracom 1000	£199

All prices include VAT and UK mainland delivery

KEYBOARDS

Schon	£57
Rainbow	£85
Schon IBM style	£99

EXPANSION BOARDS

512K Miracle Expanderam	£99
512K internal upgrade	£94
512K SQB & Mouse	£287
768K Trump Card with interface	£199
Eprom expansion board	£68

COMPUTERS

QL JS 2.3 software	£169
QL640 JS 2.3 software	£260
Thor & Z88	phone
Psion Organiser II XP	£138

SUNDRY

QL service manual	£20.00
10 Mdv cartridges	£17.50
QL dustcover	£5.00
MDV storage box	£5.50
PSION 2,3 s/tware sets	£10.00
QDOS Companion	£7.00
Assembly language programming	£9.00
Psion 2.35 software set	£20.00
Technical guide	£20.00
10 DS/DD 3.5" disks	£25.50
10 DS/DD 3.5" disks (Brand)	£34.00
JS Rom set	£27.50
Power Clean Filter	£36.00
Keyboard membrane	£6.25
QL Spares	phone
Battery clock kit	£19.50

● Phone for details of other QL products ● Credit card phone orders accepted, or send cheques, postal orders, Eurocheques to:

TK COMPUTERWARE, STONE STREET, NORTH STANFORD, ASHFORD, KENT TN25 6DF

Telex: 966676 PMFAB G. FAX: 0303 64850

VISA AMERICAN EXPRESS Access EUROCARD

UTILITY

Ron Massey looks at a new printer driver offering complete control over hard copy positioning, sizing and density. Almost entirely menu-driven, Sidewinder allows copy to be printed horizontally or sideways in a wide range of definable sizes. Other programs under review include Job Applications Manager, Desk Top Publisher and CuePrint,

Product: Sidewinder
Price: £11.50
Source: CARE Electronics, 800 St. Albans Road, Garston, Watford, Herts WD2 6NL.
Tel: 0923 672102.

For most people, printer drivers supplied with graphics programs are adequate for most hard copy needs. With varying degrees of success, the drivers can translate the QL screen image into a code adequate for the majority of printers, although there is a better chance for producing hard copy successfully if a printer is Epson-compatable.

The biggest limitation of almost all drivers is that their output is usually a fixed size, fixed screen-to-paper orientation and, except for those which offer the option of framing a picture prior to making a dump, a fixed aspect ratio.

A new product from Zitasoft, *Sidewinder,* makes most standard printer drivers obsolete. Offering the user complete control over hard copy, Sidewinder features include accurate positioning and sizing of hard copy; complete control over the density — number of passes of the printer head over the same row — and dots per head pass — to a degree, effectively controlling the amount of detail in the finished hard copy.

Almost entirely menu-driven, Sidewinder is the epitome of user-friendliness. Copy may be printed horizontally, as viewed on the screen, or sideways and in a very wide range of definable sizes.

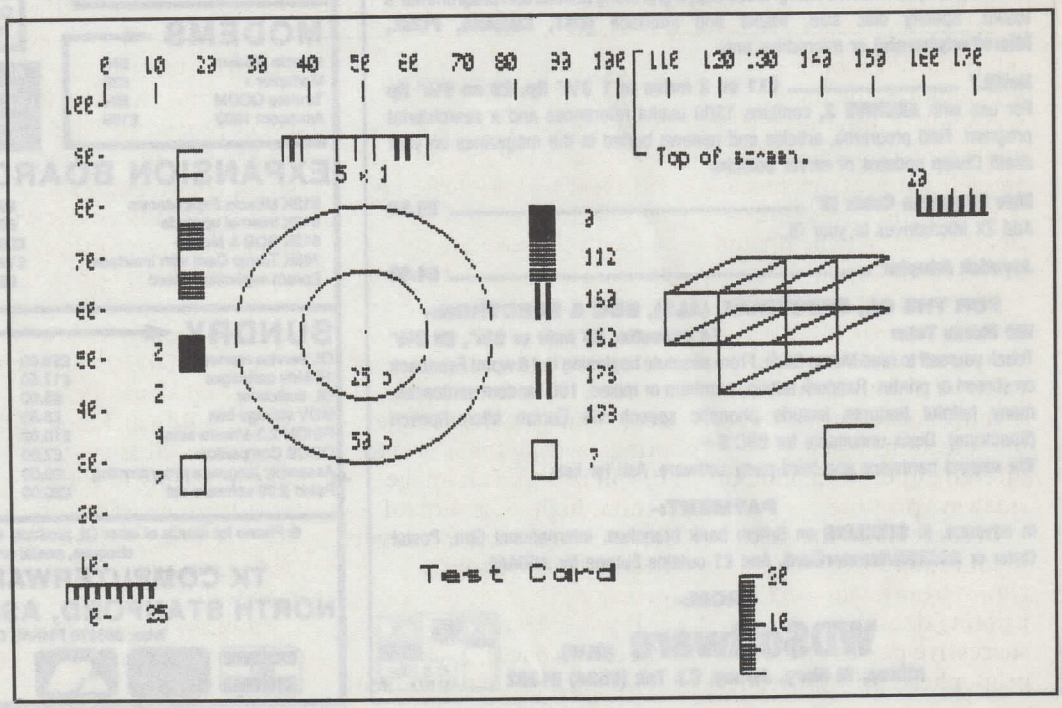
System defaults

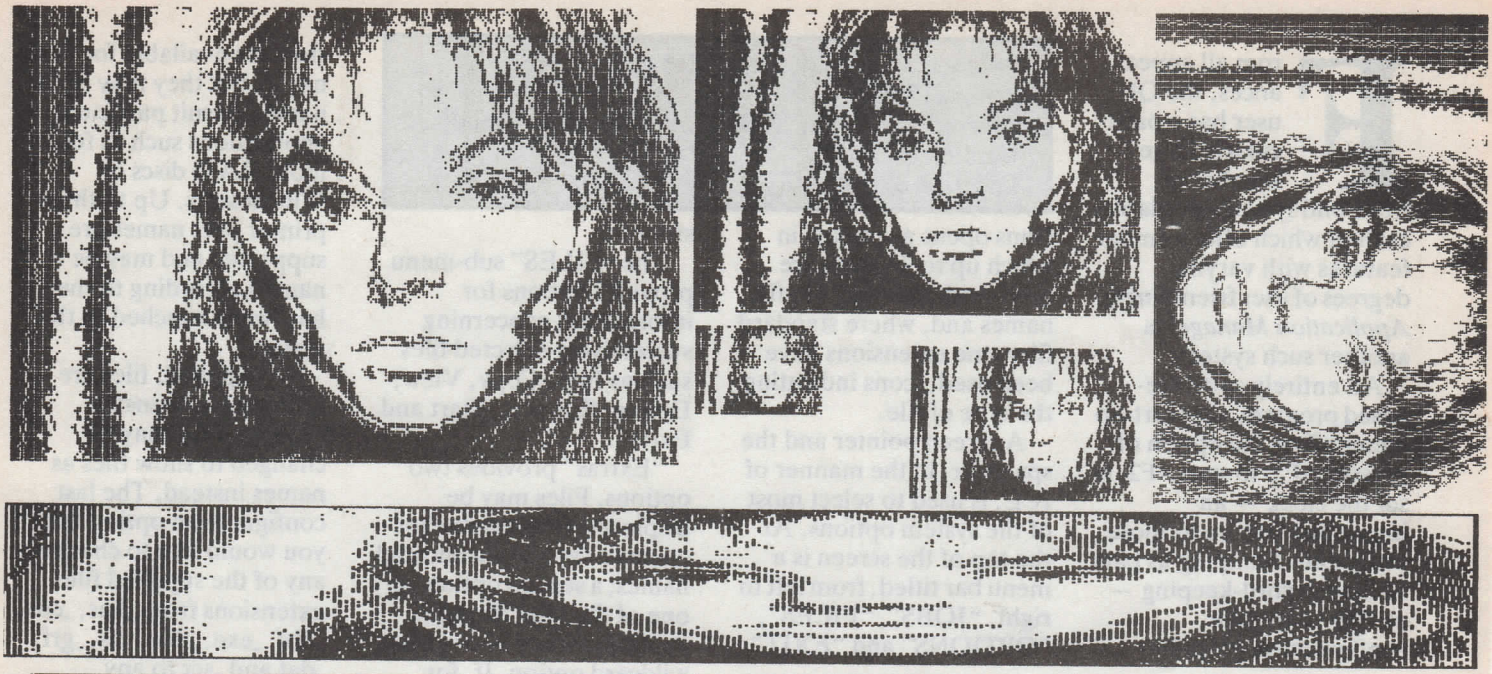
Although system defaults have been built into the program, source drives for screen images may be set from the default mdv1_ to any other drive attached to your system; baud rates may be re-set from the default 9600 to any other rate within the range of 75 to 19,200; the output port, defaulting to serl, may also be changed to any other legal QL device.

A useful range of graphics facilities is available for retouching an existing drawing with either black or white ink. The retouching brush, controlled by the cursor keys, may be expanded horizontally and vertically for either point or line retouching.

Other screen-handling features have been included which enable extensive modification of a screen printer to a dump; green and red may be re-coloured as either black or white; margins of a picture may be defined by scrolling and panning the screen image — which screen wraps, so the original margins may be restored; right and bottom margins can be re-defined by using wire cursors and the image optionally may be framed.

Television users may





Parameter \ Dump No.	1	2	3	4	5	6	7	8	9
Printer Density	1	1	1	1	1	1	1	1	3
X Density	6	1	1	2	4	6	6	6	6
Central Spaces	10	10	10	10	10	10	5	5	5
Y Size	10mm	10mm	20mm	20mm	20mm	20mm	40mm	60mm	80mm
Dump Size (mm)	11x17	11x17	20x24	20x24	20x24	20x24	40x58	59x88	79x117
Dump Time (sec)	2	1	3	4	3	7	26	44	286

Printer compatability: Nominally Epson RX80.
Printer used for these dumps: Canon PW1080A.
Printer Density Range: 1 to 3 passes.
Centralising Spaces: Number of characters from left margin.
Y Size: Approximate height of picture. Dump ratio is roughly determined from screen aspect ratio. Dump Size is surprisingly accurate.
X Density (Dots per inch):

1	480	
2	960	(Double-speed. Horizontally adj dots not printed.)
3	640	
4	720	
5	960	
6	1920	(Horizontally adj dots not printed.)

select the View option which moves the image temporarily into the centre of the screen, so that the top and left margins are visible. Selecting the Purge option clears the area beneath and to the right of the frame.

When you are satisfied with the screen version of the picture, pressing ‹F2› calls the output formatting menu. Proceeding through the selections you are offered either horizontal or sideways printing.

In succession, the next input required is your choice from a range of three printing densities; successive passes of the printer head prints two dots between each pin of the printhead. Copy density between 480 and 1,920 dots, the range available for the Epson RX80 printer, may be selected.

Central spaces, the spacing from the left-hand margin, is then set. "Y Size" is set for the required height of the printed image. As can be seen from the chart accompanying the sample prints, setting the Y size to 10 will produce an image 11mm. high. Selection of larger image sizes for larger heights is surprisingly accurate.

If you select zero "Central Spaces", optionally you may produce a stretched screen image wherein the image is not proportional but is full paper width by whatever height you select.

Users of Sidewinder will find a number of useful applications for this program. Step-and-repeat hard copy — multiple dumps from the same image and set-up — can be made, using a utility included with the system. Sidewinder accurate sizing facilities provide a means to produce any type of copy, whether for Microdrive labels or for banners.

Probably one of the most useful and versatile printer utilities yet reviewed in Utility File, I feel confident predicting that Sidewinder eventually will become an industry-standard approach for future graphic printer driver systems.

The instructions supplied with Sidewinder are down-to-earth and definitely user-friendly. Requirements for using the system are described clearly and, as its first paragraph points out, there is sufficient help built into the program "to enable you to be using the full power of the dump within 20 minutes of getting it up and running." Absolutely top marks for a superb program.

Product: Job Applications Manager
Price: £19.95
Source: Eidersoft, Hall Farm, North Ockenden, Upminster, Essex.
Tel: 0708 851099.

From all appearances, the QL user has a particularly large number of front-end systems available, each of which offers similar features with varying degrees of user friendliness. *Application Manager* is another such system.

An entirely software-based program, the start-up menu offers the option of pressing ‹F1› to start, ‹F2› to set the clock — an important feature to those using date-stamping as part of their record-keeping — or ‹F3› to run the configuration program.

UTILITY FILE

CONTINUED

Selecting ‹F1› EXECs JAM followed by a screen message that JAM is called from SuperBasic by typing "JAM".

Aimed primarily at simplifying files and current job handling, the screen is a mixture of three pull-down menus from which most of its options are chosen; and a column of five icons to the left of the screen for selecting a drive from a range of two discs, two Microdrives and one RAM disc.

Selecting one of the drive icons opens a window in which up to 24 icons are displayed, showing the file names and, where standard filename extensions have been used, icons indicating the type of file.

A screen pointer and the spacebar, in the manner of ICE, is used to select most of the system options. At the top of the screen is a menu bar titled, from left to right, "JOBS", "FILES", "OPTIONS" and "EXIT", returning you to SuperBasic.

Moving the cursor to "JOBS" in the option bar at the top left of the screen and pressing space produces a pull-down menu with a range of options for controlling jobs held in memory.

Selecting any of the five sub-options produces sub-menus from which specific jobs may be viewed or altered in terms of release, suspension — with a further sub-menu for selecting time-out — priority or stopping.

The "FILES" sub-menu provides options for information concerning successively-selected files such as Info, Copy, View, Delete, Rename, Start and Extras.

"Extras" provides two options. Files may be displayed as names/sector counts or as icons/truncated names; a second option, any one of the nine filename extensions may be used as a wildcard option. If, for example, "_bas" is chosen, all the files with that extension will be displayed in the directory window.

The "OPTIONS" menu provides access to Memory information, Notes, a traditional-format Calendar, the inevitable Calculator, and Set-up.

"Set-up allows you to alter the rate of cursor movement, either temporarily or permanently. This window was labelled "Mouse Control", so I can only assume that this was reserved for a future upgrade.

One of the features I liked is that the configuration routine allows you to set up many of the defaults to your own requirements. Only five slots are available for drive names but they may be re-named to suit particular applications such as fdk, more RAM discs or Winchesters. Up to three printer port names are supported and may be re-named according to the hardware attached to the system.

As supplied, files are indicated as icons on default. That may be changed to show files as names instead. The last configuration option asks if you would like to change any of the supplied file extensions from _bas, _cde, _dbf, _exe, _doc, _lis, _grf, _dat and _scr to any alternative three-character extension you may use. Once re-configured, the new drive names and extensions are presented in the display on default.

A surprising omission from the system, especially in the light of developments with other utilities such as *QRAM, Taskmaster* and *QWICK* is that JAM does not support any form of Hotkey; the only means of activating it is either from Basic, on the command line, or from within a program.

Although it is reasonably priced, I felt that users would find most of the slightly more expensive utilities much easier to justify and almost all include a greater range of features.

Product: The Editor
Price: £39.95
Product: BetterBasic
Price: £24.95
Product: Desk Top Publisher
Price: £69.95
Source: Digital Precision, 222 The Avenue, London E4 9SE.
Tel: 01-527 5493.
Product: CuePrint
Price: £37.95
Source: D. A. Burridge, 23 Cromwell Ave., Hammersmith, London W6 9LA.
Tel: 01-748 3437.

As successive upgrades of *The Editor* pass my desk, I wonder how much further Chas Dillon can take this already powerful program. The current 1.182 version has succeeded in increasing its processing power several more notches.

Starting with version 1.18, The Editor is compatible with *WRAM*. Version 1.181 showed some relatively minor improvements, at least when compared to the next version on. One of the improvements to 1.181 is that a new command, CD (Cursor Delay), has been impleted whereby the speed of the cursor movement is completely under user control. By entering "CD *n*", where *n* is any number between 1 and 30000, cursor movement can be controlled independently of the hardware on which Editor is running.

At the top end of the CD scale, character incrementation occurs for a held-down key about once every two seconds. Other major improvements to Editor 1.181 include extended commands for paragraph indents and left and right margins have been implemented. The modified commands, if they are not

followed by a numerical qualifier, now set the appropriate margin at the current cursor position.

Modifications

A number of modifications have been installed in version 1.1812 which provide many useful new or amended facilities:

* If a syntax error has occurred, interruption of the execution of a command file now shows, in addition to "Syntax Error", a negative line number, indicating at which line of the command file the error occurred.
* Paragraph Reform now recognises any temporary left margin.
* GOTO is now available for both line and character numbers.
* Formerly EX, MR now performs the margin release function. EX is used to EXecute command lines included in an otherwise pure data file. Future editions of The Editor may use these command lines for purposes such as setting tabs, margins and other document formatting information.

Block handling has been extended considerably so that blocks may be defined as lines, as in previous Editor columns or characters, the latter of which is almost identical to the copy command supported by Quill, *WordStar* and other word processors.

The column block is defined by upper left and lower right block start and block end positions respectively, effectively the area of the rectangle thus formed. Another new block command is BH (Block Hide). When active, the defined block is indicated by inverse INK and PAPER for each of the block command types.

Defining a block with a start at column 10, line 20 and an end at column 30, line 50, entering each of the block Type commands in succession indicates the area defined as noted for a column block; a character block is shown as noted except that all the characters between the start and end of a block are highlighted. As in previous Editors, column one, line 20 and the last column in line 50 are shown for a line block.

New prompts

Additions to the Digital Precision *BetterBasic,* also by Chas Dillon, provide new prompts for users with drive names not already included in its range of choices, structure indenting — fixed at two characters in previous versions of BB — the option of electing to have IF/ELSE/END IF in the same column or ELSE/END IF in the same column as the command structure.

Another improvement to BetterBasic is that it now asks if the Basic file under scrutiny is intended to be Supercharged, the compiler for which BetterBasic was originally written.

If a program is to be re-formatted Basic for Supercharge, sliced literals will be converted, as in previous versions of BB. Since Turbo supports sliced literals, if a program is Turbo-bound conversion will not be made.

Third revision

Multiple structures on a single line are folded automatically for Supercharge applications and are not folded, except when re-formatting has been chosen and a line exceeds its pre-defined length, when used for programs to be compiled with Turbo.

A third revised Digital Precision product, *Desk Top Publisher,* has incorporated a number of refinements among its already extensive range of commands. Method of command entry is now standardised throughout the program.

In addition to entering commands with combinations of keys, as in the version one release, you now have menus for each operation mode in which a cursor bar, moved with the vertical cursor keys, selects a command which is then called when ‹SPACE› is pressed.

Two methods

Graphic screens can now be loaded in either of two ways — as in version one, by selecting the load screen option from the main menu, you are asked if you wish to load a full screen — standard 32K screen file — or a compressed Eye-Q screen. You then select the portion of the screen you wish to have printed on a DTP page.

Version two DTP also allows you to load a 32K screen file directly into the section of the page on which you are working. With this option, however, it must be remembered that, because of the framing of the work area, image clipping will occur on the left and right sides of the picture being loaded. You should ensure that the picture area you wish to include in an edited page is positioned accordingly.

Sections of pages being edited may be saved as discrete files, in addition to saving the entire page. Some very useful graphics routines have been added to version two which allow you to draw circles, ellipses, lines and arcs directly to the current page. Boxes in version two, as in version one of DTP, are drawn at the text input stage. The last operation mode, the fill section, allows you to fill outlines with any of the graphic shapes from sprite character font A.

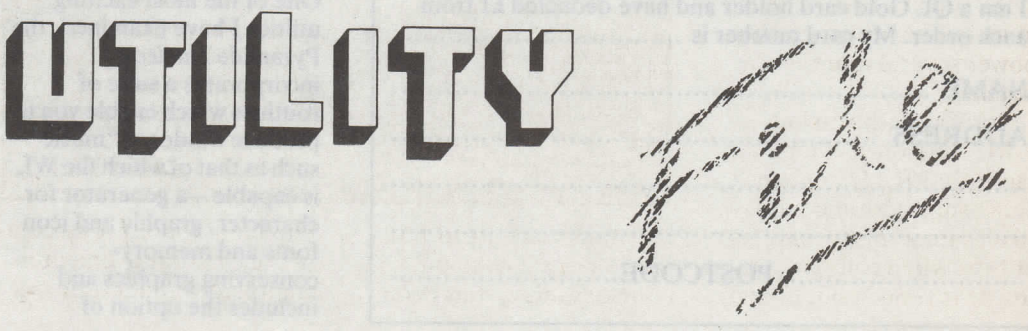
The *CuePrint* system from D. A. Burridge has been expanded considerably. A number of utilities have been added to the system, providing a means to manipulate almost any type of document requiring high-quality printing.

One of the utilities, *Patch,* enables you to intercept the output to a printer port and re-direct the stream to a drive and filename. This module is used for merging other files, usually graphic, with CuePrint documents.

A complete sub-system is provided for combining graphic files, for use with either mode 8 or mode 4, producing defined printer reverse linefeed and adding a block of text adjacent to a window automatically.

Files may be imported for use with Cueprint mail-merging modules. Output may be directed to a file instead of a printer and may be used independently of CuePrint by using the Copy_N, without a file header, to a printer port. Such files may be imported subsequently back into CuePrint to form more complex macros.

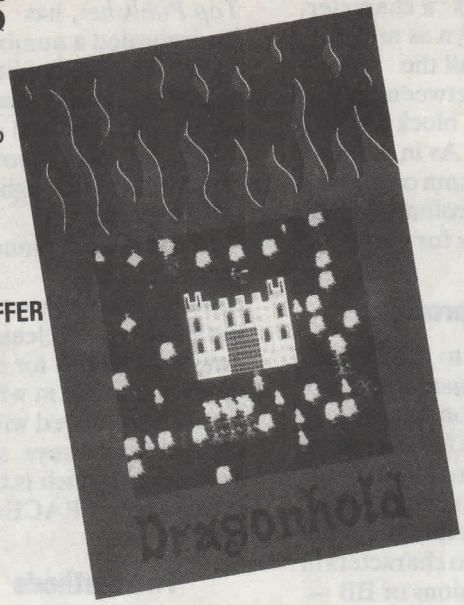
The CuePrint manual has been produced with the CuePrint system and is a good advertisement for the potential of the system.



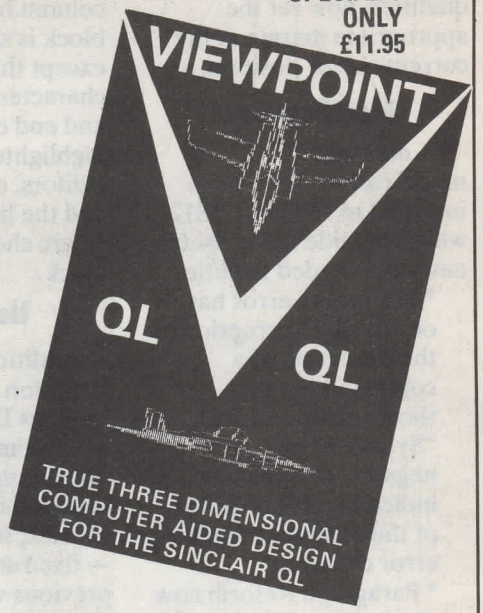
RUBICON MICROS READERS OFFER

In association with Rubicon Micros we offer readers a chance to purchase two of the foremost utility and entertainment programs currently available for the QL.

SPECIAL OFFER ONLY £15.95



SPECIAL OFFER ONLY £11.95



Both programs have received excellent reviews and these latest versions will also multi-task.

DRAGONHOLD *(MRP £19.95)*

A well crafted graphic animated adventure game set in six magic lands full of goblins, laughing dwarves and vampire bats. The arcade element gives it the edge over more conventional games.

VIEWPOINT *(MPR £14.95)*

Is written in assembler and incorporates the powerful professional features required of a 3D modelling program. These include a number of superb demonstration files, good resolution even on a low resolution screen, and the manual is exceptionally well planned.

Please send me ☐ copy(s) of Viewpoint ☐ Dragonhold ☐
I include £....
Method of payment
P/O ☐ CHEQUE ☐ (please tick).

NAME .....

.....

ADDRESS .....

.....

.....

..... POSTCODE.....

PLEASE ALLOW 28 DAYS FOR DELIVERY

To order fill in the coupon and send with your cheque/postal order
To: Rubicon Offer
Sinclair QL World
Greencoat House
Francis Street
LONDON SW1P 1DG

PYRAMIDE OFFER

EXTRA £1 OFF FOR QL GOLD MEMBERS!

There are no give-aways this month but we have something that everyone can take advantage. We have managed to persuade Pyramide to offer some of its software packages at very special prices, exclusively to readers of *Sinclair QL World*.

Wanderer, Mortville Manor and *Nucleon,* which normally cost £17 each are available for only £12. you can save £2.95 on the usual asking price for *Graphic Toolkit* which is only £13.00.

Please send me

Copies of Wanderer	@ £12 each. Total	.....
Copies of Mortville Manor	@ £12 each. Total	.....
Copies of Nucleon	@ £12 each. Total	.....
Copies of Graphic Toolkit	@£13 each. Total	.....
	Sub Total	.....
	Post and packing 75p	
	Grand total	.....

I am a QL Gold card holder and have deducted £1 from each order. My card number is .....

NAME .....

ADDRESS .....

.....

.....

..... POSTCODE.....

All the software has been reviewed in previous issues. of *Sinclair QL World* but for those who missed out, here is a little of what we had to say.

Graphic Toolkit

Very much up to the standard set by other Pyramide graphic products such as *QL Peintre* and *Nuclean,* Graphic Toolkit is an invaluable aid for providing a slick, professional finish to any program.

Nucleon

One of the most exciting utilties I have examined, the Pyramide *Nucleon* incorporates a suite of routines which enable you to produce windows, "music" – such as that of which the WL is capable – a generator for character, graphic and icon fonts and memory-conserving graphics and includes the option of generating SuperBasic programs at the end of each section for use in your programs.

Mortville Manor

Is made more enjoyable by the very good graphics and the tremendous attention to detail, for example the light which casts shadow in the bureau. The graphics greatly enhance the atmosphere of the Manor and provide every budding detective with ample opportunity to glean useful clues.

Send your completed order, together with a cheque/PO made payable to Sinclair QL World Ltd to:

Pyramide Offer
Sinclair QL World
(Merchandising)
Greencoat House
Francis Street
London SW1P 1DG.

Please allow 28 days for delivery.

DIY TOOLKIT

Each month Marcus Jefferey adds novel procedures and functions to the QL repertoire. This month he explains how to save an area of the screen to memory for later recall.

We asked you for your ideas on what routines you would like to see featured in this column. We have already had a large number of replies. Some are just too big for the space available but many of the ideas are very good and will be developed during the next few months.

One of the most requested extensions to SuperBasic is a command to save an area of the screen to memory, so that it may be loaded back at a later date. In addition to being generally useful, a command of this kind forms the basis of a multi-window package, where over-written windows may be restored at a later date.

To achieve this effect, the program this month will develop two new procedures, G_SAVE and G_LOAD, which will take the form:

G_SAVE X, Y, WIDTH, HEIGHT, ADDR

and:

G_LOAD X, Y, ADDR

where:

X, Y are the top left co-ordinates of the area to be saved

WIDTH is the width of the area

HEIGHT is the height of the area

ADDR is the start address in memory to which the screen data is to be saved

To make the commands more flexible the width and height of the screen area are saved to memory with the screen data. They are then read back by the G_LOAD command, so they do not have to be repeated in the parameter list. In addition, the X, Y co-ordinates, given in the G_SAVE command, provide a useful method of copying portions of the screen to different areas. This use is highlighted in the demonstration program.

The ADDR normally will be set up using the RESPR SuperBasic function which reserves memory in the Resident Procedure area. Bear in mind that the entire QL screen requires 32K of memory — figure one — so you will have to save that amount plus eight bytes for the width and height if you intend to use the G_SAVE command, providing a useful method of copying portions of the screen to different areas. This use is highlighted in the demonstration program.

The ADDR normally will be set up using the RESPR SuperBasic function which reserves memory in the Resident Procedure area. Bear in mind that the entire QL screen requires 32K of memory — figure one — so you will have to save that amount plus eight bytes for the width and height if you intend to G_SAVE the whole screen at any point.

The X,Y co-ordinate parameters should be given using the Pixel Co-ordinate

Figure 1

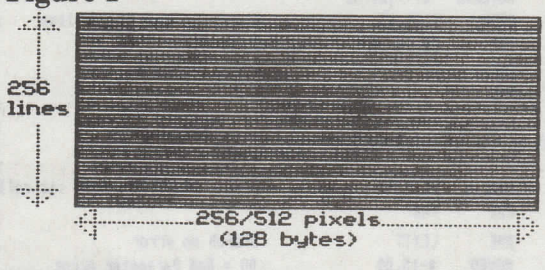


Figure 3

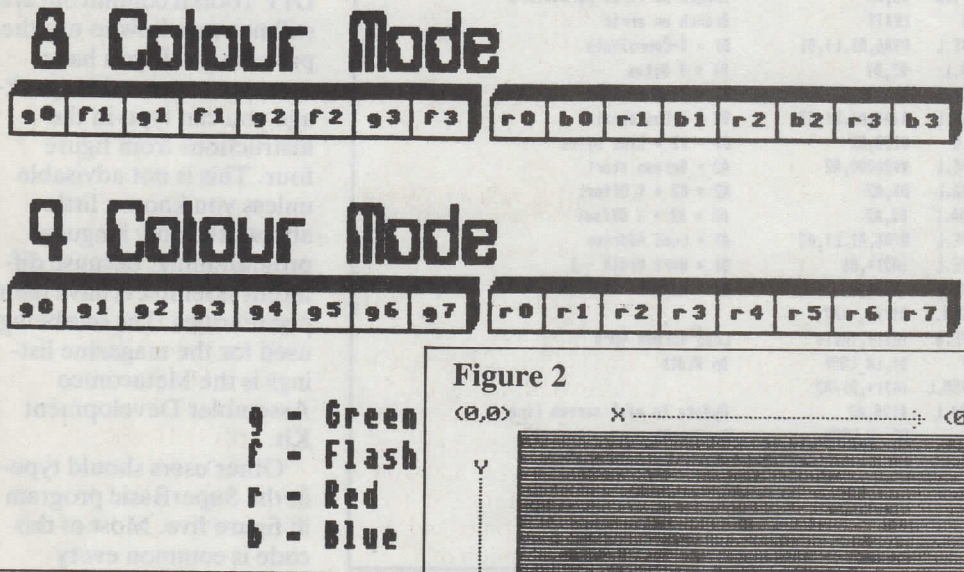


Figure 2

Figure 4

```
                                1 *
                                2 *  First add the two Procedures
                                3 *
0000' 3078 0110                 4          MOVE.W   $110,A0          $110 = BR.INIT = Add proc/func
0004' 43FA 0006                 5          LEA.L    PROC,A1          Link in procedure/function
0008' 4E90                      6          JSR      (A0)
000A' 4E75                      7          RTS
000C' 0002                      8 PROC     DC.W     2                Number of procedures
000E' 001A                      9          DC.W     SAVE-*           Relative location of G_SAVE
0010' 06                       10          DC.B     6                Length of procedure name
0011' 475F 5341 5645           11          DC.B     'G_SAVE'         Name of procedure
0018' 0074                     12          DC.W     LOAD-*           Relative location of G_LOAD
001A' 06                       13          DC.B     6                Length of procedure name
001B' 475F 4C4F 4144           14          DC.B     'G_LOAD'         Name of procedure
0022' 0000                     15          DC.W     0                End of procedures
0024' 0000                     16          DC.W     0                Number of functions
0026' 0000                     17          DC.W     0                End of functions
                               18 *
                               19 *  This is the G_SAVE procedure
                               20 *
0028' 3078 0118                21 SAVE     MOVE.W   $118,A0          $118 = CA.GTLIN = Long int collect
002C' 4E90                     22          JSR      (A0)
002E' 6600 005A                23          BNE      SEXIT            Branch on error
0032' 70F1                     24          MOVEQ    #-15,D0          D0 = Bad Parameter error
0034' 0C43 0005                25          CMPI.W   #5,D3            Should be five parameters
0038' 6600 0050                26          BNE      SEXIT            Branch on error
003C' 2236 9800                27          MOVE.L   0(A6,A1.L),D1    D1 = X-Coordinate
0040' E489                     28          LSR.L    #2,D1            D1 = X Bytes
0042' 02B1 0000 007E           29          ANDI.L   #$007E,D1        D1 = Even (0-126)
0048' 2436 9804                30          MOVE.L   4(A6,A1.L),D2    D2 = Y-Coordinate
004C' C4FC 0080                31          MULU     #128,D2          D2 = D2 * Line Bytes
0050' 247C 0002 0000           32          MOVE.L   #$20000,A2       A2 = Screen start
0056' D5C1                     33          ADDA.L   D1,A2            A2 = A2 + X Offset
0058' D5C2                     34          ADDA.L   D2,A2            A2 = A2 + Y Offset
005A' 2676 9810                35          MOVE.L   16(A6,A1.L),A3   A3 = Save Address
005E' 2236 9808                36          MOVE.L   8(A6,A1.L),D1    D1 = X-Width
0062' 5381                     37          SUBQ.L   #1,D1
0064' E689                     38          LSR.L    #3,D1
0066' 26C1                     39          MOVE.L   D1,(A3)+         Store Word Width - 1
0068' 2436 980C                40          MOVE.L   12(A6,A1.L),D2   D2 = Y-Height
006C' 5382                     41          SUBQ.L   #1,D2
006E' 26C2                     42          MOVE.L   D2,(A3)+         Store Height - 1
0070' 48E7 4020                43 SH_LOOP  MOVEM.L  D1/A2,-(A7)
0074' 36DA                     44 SW_LOOP  MOVE.W   (A2)+,(A3)+      Save screen word
0076' 51C9 FFFC                45          DBF      D1,SW_LOOP       Do Width
007A' 4CDF 0402                46          MOVEM.L  (A7)+,D1/A2
007E' D5FC 0000 0080           47          ADDA.L   #128,A2          Update to next screen line
0084' 51CA FFEA                48          DBF      D2,SH_LOOP       Do Height
0088' 4280                     49          CLR.L    D0               Signal no error
008A' 4E75                     50 SEXIT    RTS
                               51 *
                               52 *  This is the G_LOAD procedure
                               53 *
008C' 3078 0118                54 LOAD     MOVE.W   $118,A0          $118 = CA.GTLIN = Long int collect
0090' 4E90                     55          JSR      (A0)
0092' 6600 004C                56          BNE      LEXIT            Branch on error
0096' 70F1                     57          MOVEQ    #-15,D0          D0 = Bad Parameter error
0098' 0C43 0003                58          CMPI.W   #3,D3            Should be three parameters
009C' 6600 0042                59          BNE      LEXIT            Branch on error
00A0' 2236 9800                60          MOVE.L   0(A6,A1.L),D1    D1 = X-Coordinate
00A4' E489                     61          LSR.L    #2,D1            D1 = X Bytes
00A6' 02B1 0000 007E           62          ANDI.L   #$007E,D1        D1 = Even (0-126)
00AC' 2436 9804                63          MOVE.L   4(A6,A1.L),D2    D2 = Y-Coordinate
00B0' C4FC 0080                64          MULU     #128,D2          D2 = D2 * Line Bytes
00B4' 247C 0002 0000           65          MOVE.L   #$20000,A2       A2 = Screen start
00BA' D5C1                     66          ADDA.L   D1,A2            A2 = A2 + X Offset
00BC' D5C2                     67          ADDA.L   D2,A2            A2 = A2 + Y Offset
00BE' 2676 9808                68          MOVE.L   8(A6,A1.L),A3    A3 = Load Address
00C2' 221B                     69          MOVE.L   (A3)+,D1         D1 = Word Width - 1
00C4' 241B                     70          MOVE.L   (A3)+,D2         D2 = Height - 1
00C6' 48E7 4020                71 LH_LOOP  MOVEM.L  D1/A2,-(A7)
00CA' 34DB                     72 LW_LOOP  MOVE.W   (A3)+,(A2)+      Load screen word
00CC' 51C9 FFFC                73          DBF      D1,LW_LOOP       Do Width
00D0' 4CDF 0402                74          MOVEM.L  (A7)+,D1/A2
00D4' D5FC 0000 0080           75          ADDA.L   #128,A2          Update to next screen line
00DA' 51CA FFEA                76          DBF      D2,LH_LOOP       Do Height
00DE' 4280                     77          CLR.L    D0               Signal no error
00E0' 4E75                     78 LEXIT    RTS
                               79
                               80          END
```

System, as shown in figure two. It is the same system used by the SuperBasic WINDOW command, making it relatively easy to set up windows and then save them to memory.

Figure three shows the method by which screen pixels are stored in screen memory. Whether in four- or eight-colour mode, the screen is divided effectively into Word-signed (two bytes) sections. It would be possible to write a routine which would save any number of horizontal pixels from any point but it is much easier if the X co-ordinate and width are multiples of eight.

This has been done in the procedures presented. The X start co-ordinate is divided by four and made even, giving a bytes figure in the range 0 to 126. Since this value is rounded down, it would make sense to round the width upwards.

Because of the way the routine works, using the assembler DBF instruction, both the width and the height need to be one fewer than the number of times we want the loops to operate. Consequently rather than adding the value seven to the width, then truncating it, we have subtracted one and truncated, giving the same answer. That can be seen by following the assembly code in figure four.

We have received a number of letters from people interested in the DIY Toolkit column but are still not sure how to use the procedures. If you have access to an assembler package you can type-in the instructions from figure four. This is not advisable unless you know a little about assembly language programming, because different assemblers have their peculiarities. The one being used for the magazine listings is the Metacomco Assembler Development Kit.

Other users should type-in the SuperBasic program in figure five. Most of this code is common every

DIY TOOLKIT

month, with just the DATA statements changing, so you will have to type it only once. If you examine the contents of the DATA statements you will find that they match the hexadecimal numbers in the assembly language listing, though you have to be careful. The assembler inserts the occasional zero byte automatically to maintain word boundaries, though they do not always show in the listing.

On running the SuperBasic Hex Loader program, if all goes well you will be asked for a filename, to which you should type something along the lines:

mdv1_screens_obj

If you are not asked this, you will probably get a message telling you the checksum is incorrect. The checksum is just a running total kept by the computer as it transfers the hex data into memory. If you are told that it is incorrect, it means that you have made a mistake when typing the data and you should re-check those lines.

Now that the machine code is safely stored on Microdrive, the extensions are ready to use. To do this you need to reserve some space in memory, load the machine code, then call it to set up the additions. This is done easily with the following commands:

```
a = RESPR (250)
LBYTES mdv1_screens_obj,a
CALL a
```

Everything is now ready and can be checked by typing and running the demonstration program given in figure six. When you are typing the program, use lower-case to type the words 'g_save' and 'g_load'. They should appear in upper-case, i.e., 'G_SAVE' and 'G_LOAD', in the listing, indicating that the additional procedures are definitely in memory.

Running the demonstration program will set up a small window with a border and print "QL World" in it. It will then save this to memory and re-load it continually at random screen locations. The program contains an infinite REPEAT loop, so the only way to stop it is by pressing ‹CTRL› and ‹SPACE›.

Figure 5

```
100 REMark : Sinclair QL World
110 REMark : ** HEX  LOADER **
120 :
130 CLS
140 RESTORE
150 READ space
160 start = RESPR(space)
170 PRINT"Loading Hex..." : hex_load start
180 INPUT"Save to file...";f$
190 SBYTES f$,start,byte
200 STOP
210 :
220 :
230 DEFine PROCedure hex_load(start)
240 :
250    DEFine FuNction decimal(x)
260    RETurn CODE(h$(x))-48-7*(h$(x)>"9")
270    END DEFine decimal
280 :
290 byte = 0 : checksum = 0
300 REPeat load_hex_digits
310    READ h$
320    IF h$="*" THEN EXIT load_hex_digits
330    IF LEN(h$)<>2*INT(LEN(h$)/2) THEN
340       PRINT"Odd number of hex digits in: ";h$
350       STOP
360    END IF
370    FOR b = 1 TO LEN(h$) STEP 2
380       hb = decimal(b) : lb = decimal(b+1)
390       IF hb<0 OR hb>15 OR lb<0 OR lb>15 THEN
400          PRINT"Illegal hex digit in: ";h$
410          STOP
420       END IF
430       POKE start+byte,16*hb+lb
440       checksum = checksum + 16*hb + lb
450       byte = byte + 1
460    END FOR b
470 END REPeat load_hex_digits
475 PRINT checksum
480 READ check
490 IF check <> checksum THEN
500    PRINT"Checksum incorrect. Recheck data."
510    STOP
520 ELSE
530    PRINT"Checksum is correct"
540    PRINT"Data entered at: ";start
550 END IF
560 END DEFine hex_load
570 :
580 REMark : Space requirements for the machine co
de
590 DATA 250
600 :
610 REMark : Machine code data
620 DATA "3078011043FA0006","4E904E750002001A"
630 DATA "06475F5341564500","007406475F4C4F41"
640 DATA "4400000000000000","307801184E906600"
650 DATA "005A70F10C430005","6600005022369800"
660 DATA "E48902810000007E","24369804C4FC0080"
670 DATA "247C00020000D5C1","D5C2267698102236"
680 DATA "98085381E68926C1","2436980C538226C2"
690 DATA "48E7402036DA51C9","FFFC4CDF0402D5FC"
700 DATA "0000008051CAFFEA","42804E7530780118"
710 DATA "4E906600004C70F1","0C43000366000042"
720 DATA "22369800E4890281","0000007E24369804"
730 DATA "C4FC0080247C0002","0000D5C1D5C22676"
740 DATA "9808221B241B48E7","402034DB51C9FFFC"
750 DATA "4CDF0402D5FC0000","008051CAFFEA4280"
760 DATA "4E75","*",19302
```

The demonstration program in Figure 6 will set up a small window and print "QL World" in it.

Figure 6

```
100 b=RESPR(3000)
110 WINDOW 152,32,40,40
120 BORDER 6,2,4
130 CSIZE 3,1
140 PRINT "QL World"
150 G_SAVE 40,40,152,32,b
160 REPeat load_screen
170    x = 2 * RND(0 TO 180)
180    y = RND(0 TO 224)
190    G_LOAD x,y,b
200 END REPeat load_screen
```

Steve Sutton compares the Q_Liberator and Turbo SuperBasic compilers. On test, Turbo 1.13 and Q_Liberator 3.11

Compilers

You might be forgiven for concluding that the QL is similar to most other small computer systems available. It has a powerful implementation of the Basic programming language, plenty of memory and other features to commend it but, unless you are willing to write programs in machine code, there is no inherent way you can use one of its most useful and unusual features, multi-tasking.

Using SuperBasic programs can also be a time-consuming experience; the interpreter is far from being the fastest implementation of Basic and any program either does a great deal of number-crunching or is more than a few hundred lines long is likely to become painfully slow in operation.

One way to make use of the full capabilities of the QL is to buy a compiler for one of the numerous programming languages available such as Pascal or Forth. This will certainly allow programs to run much faster than using SuperBasic and will also allow the multi-tasking features to be used. The disadvantage is that programs cannot be tested under the interpreter as they are developed and the programmer must become familiar with yet another language.

Simple way

The simple way round these disadvantages is to use a compiler for SuperBasic. It allows programs to be developed interactively using the interpreter and then, once they are working correctly, compiled, allowing them to be multi-tasked and also to run much faster.

There are two compilers available for SuperBasic — three if you count SuperCharge, the forerunner of Turbo — Q_Liberator from Liberation Software and Turbo from Digital Precision. Both products will allow a SuperBasic program to be compiled and each provides numerous other features.

A compiler works in a fundamentally different way from an interpreter. Before a program can be run in a compiled form, each statement in the source is examined by the compiler and converted to a form more suitable for the computer to understand. This process, compilation, must be done before the program can be used but, provided the program does not need to be altered, it is unnecessary to repeat it again. The compiled form of the program is saved as a separate file and may be run as often as required.

The same program, if run under an interpreter, is treated very differently. For example, when the RUN command is used, the SuperBasic interpreter starts to examine each line of the program in turn and convert it, statement by statement, to a form suitable for the QL 68000 processor to work on. The conversion process is similar to that done by a compiler but, unlike the compiler, the results of the conversion are not stored permanently. Instead, the compiled code for each statement is executed as soon as it is produced, so if the same line of the program is repeated as part of a FOR loop, the time-consuming conversion process must be repeated again and again. This repeated re-compiling of the program is one reason interpreters are relatively slow.

This description is, of course, a very simplified view. The two SuperBasic compilers, while designed to do a similar job, go about it in different ways. Turbo converts the source into a form which is executable directly by the 68000 processor. The Q_Liberator compiles the source into an intermediate form which is then used by a run-time system in much the same way the interpreter runs the original.

This process is made much more efficient than interpreting the program by the design of the intermediate code stage. By arranging for most of the time-consuming parts of the compiling process to be handled by the conversion to the intermediate form, the resulting program still runs very much faster than it would with SuperBasic.

Not important

The way each compiler works in detail is not important when using them. A brief glance at the advertising associated with each product will reveal that Q_Liberator seems to have compatibility with SuperBasic as one of its most important features, while Turbo takes the execution speed as one of its main selling-points. It is important to keep those different objectives in mind when comparing the two products.

Both compilers use a similar front panel to control how a program is to be compiled. In each case, the chosen SuperBasic program, which should, of course be fully-tested under the interpreter, is loaded and then a variety of options is available to allow it to be compiled. Q_Liberator allows a compilation to be started by the use of a single command such as:

LIBERATE MDV1_TEST,

That single line will cause the compiler to be invoked using a set of default options. The resulting compiled version will be called *TEST_OBJ* and will be saved on MDV1. The compiled program can be run using the standard SuperBasic commands EXEC or EXEC_W or, alternatively, one of the extra procedures made available by the system can be used. For example:

QX MDV1_TEST

will perform a similar function to EXEC MDV1_TEST_OBJ. Turbo can be used in much the same kind of way. The use of the command:

CHARGE MDV1_TEST_TASK

will start the compiler and then it is necessary only to press the spacebar to proceed with the compilation. Turbo uses four passes of the source program; it reads it through four times to complete the compilation, while Q_Liberator works with two passes. In terms of the total time taken to compile, however, I found little difference between them.

Turbo, as part of the Turbo Toolkit, also provides numerous extra commands to enable compiled programs to be run. They are of most use when more advanced features of Turbo are made use of. The EXEC command also has its counterpart with Turbo in the form of EXECUTE. This can be used in exactly the same way as EXEC but provides additional features, such as the ability to specify at what priority a compiled program is to run and also to allow communication pipes to be established between different compiled programs.

The front panel of each program allows various options to be selected which will affect how a compilation is performed. Each system allows internal line numbers to be suppressed which can reduce the size of the compiled program substantially. The documentation on each compiler is very clear and I had no difficulty understanding what each option was for.

A particular feature with Q_Liberator is the provision for on-line help. If you

cannot remember how a particular selection works, moving the cursor over the required option and pressing F1 will call-up a full page of help text; this certainly avoided numerous trips into the manual when first getting used to the system.

One other additional feature with the Q_Liberator front panel is the ability to use QRAM for the selection and control of options. This refinement adds appreciably to the user-friendliness of the compiler.

First comparison

Before attempting to compare how programs compiled by the two systems run, it is important to compare how well each compiler can cope with different SuperBasic programs. To this end, 11 SuperBasic programs were tested with each. The programs are all by different authors and so present a variety of programming styles; two of them are more than 1,000 lines long and they were all chosen to cover a variety of applications, ranging from an assembler to a graphics drawing package. The results of attempting compilations of each program are summarised in table one.

The first point to note is that neither of the compilers was completely able to handle all 11 programs without some modification being made. Two of the programs, which will remain nameless, contained structures which defeated both. This is probably more of a reflection on how SuperBasic is often able apparently to run badly-structured code correctly. I could not make sense of some of the sections of these particular programs.

More warnings

An important observation resulting from the 11 test compilations is that, generally, Turbo produced many more warning messages than Q_Liberator. Any error found by either program will result in no output file being produced, while a warning is used either to draw attention to a potential problem or to indicate where the compiler has effected some correction to the code.

This probably indicates that Turbo is less willing to tolerate unusual programming techniques then Q_Liberator and seems to attempt more corrections to the source code. Both compilers can correct some incorrect structures automatically; either will detect a conditional END DEFINE and convert it to a RETURN statement.

Of the 11 programs tested, Turbo was able to compile nine, while Q_Liberator managed eight without corrections being applied. Of those compiled successfully, the Liberated programs all ran, while two of the Turbo versions gave problems.

At this point it should be mentioned that Turbo imposes some restrictions on the use of certain features of SuperBasic. Of the ones of which I am aware, the most important are:

Incomplete emulation of the SuperBasic parameter-passing mechanism. This will prevent a procedure operating on different data types.

Calculated line number references are not supported. For example, RESTORE n*5.

Expressions are not allowed in DATA statements.

Some features of SuperBasic are not supported by either compiler. The use of RENUM, MERGE and several other interpreter-specific features are not applicable once compiled.

Program length

Generally, the length of programs is about the same between compilers, differences of no more than 1.5K in length being noticed with any of the tested programs. The effective length of the Liberated versions can be reduced by about 10K for each program by excluding the run-time system from the compiled code. If that is done, a copy of the run-time must be loaded into memory before a program is run and there is little point in it if a program is designed to run by itself. If more than one program is to be run, however, each may share the same run-time code, saving about 10K for each additional program.

A great deal of emphasis is often placed on how fast a compiled program will run. When comparing the SuperBasic compilers it is instructive to compare each compiler not only with the other but also with the interpreted performance. Taking simple test programs designed to examine specific performance features gave results as summarised in table two.

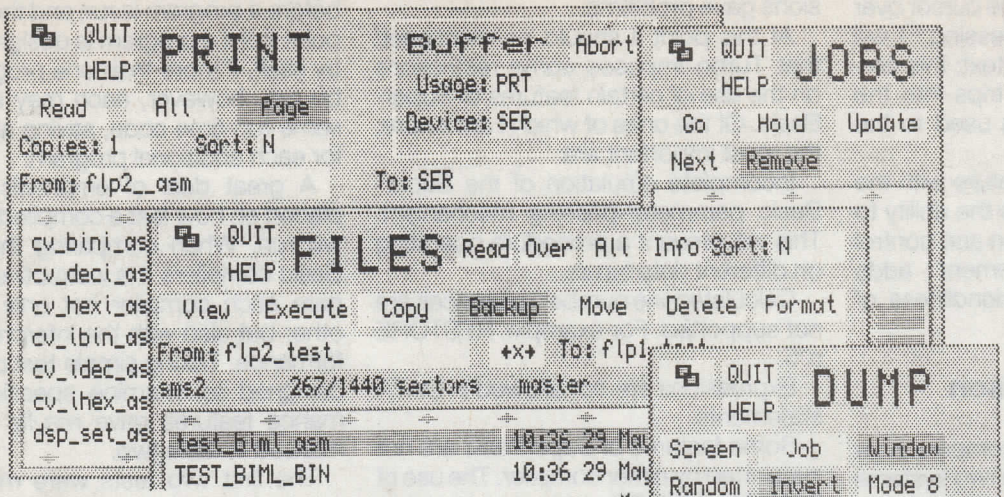
The first two tests were made with short programs, similar to standard benchtests. The numbers indicate how much faster the programs ran, once compiled, compared to SuperBasic. The most dramatic improvement in speed is achieved with integer arithmetic Turbo manages a 13:1 increase in speed over SuperBasic while Q_Liberator also achieves a substantial improvement of 8.7 times.

Both compilers have options which can alter the performance of compiled code. By and large I have tried to use them in as near the same way as is pos-

Lines and length	Description	Compile time (seconds) TURBO	and comments	Q_LIBERATOR	
1903 77K	Graphics drawutility	372. 9 warning messages	yes	333.1 warning	yes
707/26K	Compiler	123. 3 warnings	yes	185.	yes
232/7K	Board game	48. 7 warnings	yes	40.	yes
1762/60K	Assembler	375. 21 warnings	yes	310.1 warning	yes
85/2.6K	Arcade game	26. 2 warnings	yes	28. 2 errors	no
47/1.1K	Copy utility	12.	no	13.	yes
362/11K	Graphics game	75. 47 warnings, 16 errors	no	77. 14 warnings, 8 errors	no
54/1.7K	Graphics utility	18. 3 warnings	yes	16.	yes
196/5.6	Arcade game	44. 2 warnings	yes	40.	yes
211/8.8K	Drawing aid	90. 12 warnings	no	64. 2 warnings	yes
692/38K	Teaching utility	136. Compiler aborted	no	253. 1 error	no

Table 1. These 11 programs were compiled using each compiler in a similar configuration. The compiled code was directed to RAM disc in each case while the compilers were run from floppy disc. Lengths indicate number of SuperBasic lines and approximate length of the source code, while the compilation times are given in seconds. Compiled programs are considered to have failed to run if errors occur during compilation. Some which compiled successfully also failed to run correctly.

Compilers



QJUMP
24 King Street
Rampton
Cambs
CB4 4QD

QJUMP SuperToolkit II. The "clear market leader" (QL World) in Toolkit ROMs. Why buy separate keydefine programs, job control utilities, default directory utilities etc. when QTK II has them all and more.

QTK II	£29.90 (£26.00 export)	On ROM / Mdv / 3½ / 5¼

Please add £1.00 post and packing (UK) or £2.00 (Europe). Payment by UK cheque or PO. Or direct transfer to Barclays Bank, Benet Street, Cambridge. Code 20-17-19, account 30744557. Or Access / Mastercard / Eurocard / Visa. Or Eurocheque in Pounds Sterling (+£200) with the card number on the reverse. International Girocheque in Pounds Sterling. Personal cheques drawn on a non-UK bank should be in local currency and should include an additional £3 to cover the bank negotiation fee.

QRAM uses the QJUMP Pointer Interface which is the new standard in multi-windowing multi-tasking environments. It gives you mouse, cursor key or single keystroke control of your QL. Only QJUMP brings you a completely new environment to your expanded QL, providing non-destructive windows for any well written program (and most badly written ones too!) no menus to set up, no silly limitations, just CTRL C to switch between programs.

QRAM provides Job and File maintenance as well as a superfast RAM disk, printer buffer, print spooler and an amazingly flexible screen dump.

QIMI is the new Internal Mouse Interface from QJUMP. A mouse interface which leaves all of the QL's expansion facilities free for other peripherals. QIMI fits right inside the QL. Tools are supplied. QIMI may not be suitable for USA or German QLs. QIMI is compatible with the ATARI mouse but QJUMP can supply a higher quality mouse.

QRAM	£29.90 (£26.00 export)	Available on Mdv / 3½ / 5¼
QIMI	£29.90 (£26.00 export)	
QIMI+mouse	£69.90 (£60.80 export)	
QRAM+QIMI	£54.90 (£47.75 export)	
QRAM+QIMI+mouse	£89.90 (£78.00 export)	

QMON II Monitor/Debugger with single line assembler/disassembler, conditional breakpoints, trace and backtrace, search etc. "Das absolute Spitzenproduct" (68000er) can we say more?

QMON II	£19.95 (£17.35 export)	Available on Mdv / 3½ / 5¼

QEP III The "Rolls Royce" (Sinclair User) of EPROM programmers. Many QLs have been bought just to use QEP III. Programs EPROMs 2716 through to 27512. Too many facilities to describe here.

QEP III	£129.95 (£113.00 export)	

QFLP floppy disk driver - this you should not need. QJUMP can supply floppy disk drivers for those who have been left with unsupported floppy disk systems. Available for Medic (gets round many of the faults on some of the early Medic disk drives) and MicroPeripherals, version 3 or version 5 (provides compatibility with the rest of the QL world and improves performance).

QFLP	£14.95 (£13.00 export)	Available for Medic MP3 MP5

TRANSFORM LTD. Computer Accessories

MAIL LIST £24.95

Allows you to retrieve names & addresses from within ARCHIVE. Other features include single key functions, prints address labels, exports files to Quill also compatible with our stock control.

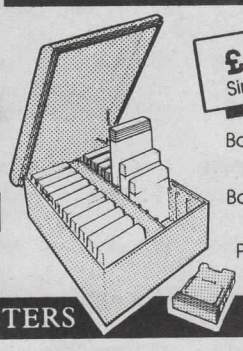
STOCK CONTROL £24.95

Keeps records of all stock items including retail price, trade price, minimum order level, items on order & many other features.

QL SWITCH £14.95

Allows users with extra memory to run Quill, Archive, Abacus & Easel at the same time or two Archive programs & Quill.

MICRODRIVE STORAGE BOX



£4.95 Single Box

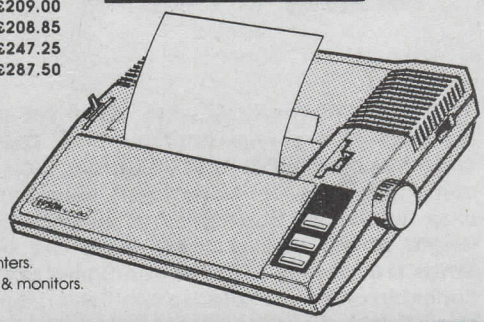
Box + 10 cartridges **£22.45**

Box + 20 cartridges **£39.95**

Please add **£1.30** p&p

PRINTERS

LX 86 215·00 + vat



DOT MATRIX

Citizen 120D	£209.00
Seikosha SP180 A	£208.85
Epson LX 86	£247.25
Taxan Kaga	£287.50

DAISYWHEEL

Brother HR20 daisy wheel	£495.00

ACCESSORIES

LX tractor feed	£20.70

We also stock ribbons for the above printers.
Please add **£5.75** for delivery of printers & monitors.

PC FOUR £79.95

A cut-down version of Xchange as supplied on the QL but with the added facility of 65,535 entries in Archive, 999 lines in Abacus & mail merge plus key define built into Quill.

QL to PC **£29.95**

PSION XCHANGE

ARCHIVE DATABASE

Multifile relational database. Only memory restricts the number of files in use at one time. Variable length records with a maximum of 255 characters per field, 255 fields per record, 65535 records per file. Command language allows the user to define procedures using a powerful editor. Flexible multifile indices provide fast access and a form designer for screen and printer input and output.

QUILL WORDPROCESSOR

Text and page formatting, underlining, bold, italic, superscript, subscript, justification, tabs, line spacing, page size, page headers, footers and numbering. On screen editing, fast page access, cut and paste, block copy, block move, insert/ overwrite modes, search and find, search and replace, key define and mail merge. Document size only restricted by disk capacity, background printing allows the user to print documents while working on others.

£295 INC. VAT

EASEL BUSINESS GRAPHICS

3D Business graphics including bar, line and pie charts, no limit to the number of values or numbers or sets of figures. Automatic rescaling, user define colours, text annotation of graphs at any pixel point. Intelligent evaluation of data formulae or manipulation of existing data. Images stored in internal format; monitor resolution is the only limit on final image quality. Accepts data from spreadsheet, database, etc.

ABACUS SPREADSHEET

Expansive spreadsheet with 999 rows, 255 columns, virtual memory system buffers to disk only when full, facilitating very large spreadsheets. Efficiency memory used only for full cells, each cell only uses 14 bytes. Variable width cells, block copy, block move, insert/ delete columns and rows, merge and consolidation of data from different spreadsheets. Accurate to 15 significant figures, full range of mathematical, scientific, date, average, ASCII and DIF conversion.

PC FOUR £79.35

Access

TRANSFORM LTD (DEPT. QL) 01-658 6350

Mail order only. 24 West Oak, Beckenham, Kent. BR3 2EZ

VISA

sible. An indication of the effect of selecting the SPEED option with Turbo is included with some of the tests. The figures show that useful improvements in speed can be achieved; the improvement over SuperBasic of the integer test is increased to 19 times by selecting this option. Further improvements can be achieved with each compiler by the use of additional features, such as arranging for FOR loops to use an integer index, which is not possible with interpreted SuperBasic.

The other tests show varying smaller improvements over SuperBasic timings. Certain operations will not benefit at all by being compiled. The writing of text to the screen is a good example; here the speed of the operation is governed by the QL hardware and machine code routines within Qdos.

Overall improvements in speed of a given program with either compiler with depend on what the program does. Applications which are highly I/O bound, either to the screen or a storage device, are less likely to run very much more efficiently when compiled. An interesting example of a program which includes a mixture of mathematical computations, logical decisions and also a fair bit of I/O is the board game used in the compilation tests.

Board game

This program will play a well-known board game against the player and the timing figures included in table two indicate that both compiled versions achieve useful speed improvements over SuperBasic. From the point of view of playing the game, the improvement of 3.3 times achieved by Q_Liberator makes all the difference between it being boring to providing an apparently more able, lively opponent. The Turbo performance is clearly even better.

Improved speed

Greater improvements in speed are possible for programs which are well-designed. Each compiler has demonstration programs designed to show various features. Taking the program supplied with Turbo first, it has a routine to calculate prime numbers. When compiled by Turbo, it runs 16.5 times faster than SuperBasic; an improvement of a more modest 4.8 times is achieved by Q_Liberator.

Redressing the balance, Q_Liberator has a sorting program supplied with it. It is a general-purpose routine capable of sorting almost any data type, integers, floating point or text. When Liberated, this routine runs 11.6 times faster than SuperBasic but a direct comparison with Turbo is not possible unless the program is modified, since one of the restrictions imposed by this compiler prevents it using one procedure to handle multiple data types.

Programs which are mostly CPU bound, such as the assembler, benefit most from speed improvements. It is with this type of program that the extra speed obtainable by Turbo is most useful.

Both compilers produce programs which can be multi-tasked, a process which allows many exciting possibilities. A simple application would be a utility program to copy files. Such a utility is able to work in the background, leaving SuperBasic or other programs still available.

Error-trapping

More advanced features offered by each compiler include error-trapping and the facility to set up communication routes or pipes between compiled programs. This allows large program applications to be split into two or more separate programs. It might be of benefit in allowing large programs which otherwise would not fit into an unexpanded QL to be run. Parts not needed at a given point can be unloaded.

Those features are simpler to use with Q_Liberator, which allows a much more flexible approach than Turbo. The latter requires that all the separate parts are loaded at one time. It is much more difficult to test the system. Q_Liberator, on the other hand, allows modules to be loaded separately as required and, in addition, allows such modules to communicate with SuperBasic or even be called as if they were procedures in it. This feature will allow libraries of compiled program modules to be built and linked to the QL as required.

Turbo includes several corrections to known bugs or shortcomings in SuperBasic. Owners of early ROMs are limited to input line lengths of 128 characters. By and large, those restrictions are duplicated by Q_Liberator, as are some other problems.

Turbo certainly produces faster code then Q_Liberator, typically by a factor of two or three, but this is partly at the expense of reduced flexibility and also some restrictions of valid SuperBasic. If speed is most important, provided you can accept the need to learn a few Turbo peculiarities it is the best option. Where speed is not a restriction, the extra facilities and virtually complete compatibility of Q_Liberator make it the better choice, especially if existing programs of doubtful construction are to be compiled.

Both compilers achieve a remarkable degree of compatibility with SuperBasic and, in addition, extend the capabilities of the language to utilise the capabilities of the QL fully. It is beyond the scope of this comparison fully to do justice to either but I have no hesitation in recommending both.

TEST	TURBO	Q_ LIBERATOR
Integer arithmetic	13.1 (19)	8.7
Floating point arithmetic	5.0 (5.4)	2.9
Trig/Log functions	1.4	1.6
Writing TEXT to the screen	1	1
Empty FOR loop, as used in above	9.2	7.9
TURBO demonstration, calculating prime numbers.	16.5	4.7
Q_LIBERATOR DEMO_SORT program	Can't compile	11.6
Independent prime number generator	5.1 (5.4)	4.8
Test running of the assembler program from Table 1	9.3	3.9
Test running of the board game from Table 1	5.7	3.3

Table 2. Comparison of the execution speed of various compiled programs. The figures quoted are speeds relative to SuperBasic; figures in brackets for Turbo refer to the user of the SPEED optimisation, generating in-line code.

Compilers

ORDER FORM

No. of programs required	☐	Total cost of programs	£ ....
No. of drives required	☐	Drive cost (£2 each)	£ ....
No. of drives sent	☐	Sub-total	£ ....
Program ID Numbers	☐☐☐☐☐☐☐☐☐	Postage & packing	£0.75
		+ 15% VAT	£ ....
		TOTAL	£ ....
QL GOLD CARD NUMBER (IF ANY)	☐☐☐☐☐☐	— 10% DISCOUNT FOR QL GOLD CARD HOLDER	£ ....
		TOTAL TO BE SENT	£ ....

Name ____________________

Address ____________________

____

Please copy on to Microdrive the programs above which I have indicated by their ID numbers. I enclose a cheque/PO to the value of £............ made payable to *Sinclair QL World*. I understand that *Sinclair QL World* undertakes to supply only these programs — copied on to Microdrive — and accepts no liability for their operation as defined by the author. Neither can *Sinclair QL World* supply additional information about any of the programs other than that originally printed or supplied. Any article reprits must be ordered and paid for separately at £2 each inclusive of post and packing — £3 overseas.

Please send order form to : Mike McKenzie, PO Box 74, Tonbridge, TN12 6DW. Tel: 089 283 4783.

PSION SOLUTIONS

Ron Massey attempts to answer more readers' queries on transferring programs with disc-based systems and on version 2.3 Psion programs.

A letter from Ian Stratton states that, having upgraded to the CST RAM expansion, disc drives and Psion version 2.3 programs, he is experiencing difficulties running one or more on his QL.

When initialised either by pressing ‹F1› or typing "1run flp1_boot", he states: "The drive runs briefly and then stops. ...the machine appears to hang for up to two minutes and, with luck, the program suddenly loads."

When going from one program to another, after re-set, "...the machine hangs again, this time indefinitely. If I revert to my old version 2.0 disc or Microdrive copies, this problem does not occur."

Since Stratton states that he is using a QL bought shortly after its launch, my initial reaction was that his QL may be suffering from power loss resulting from the upgrade add-ons and when the machine had been in use for a time.

Common to a large number of QLs, this problem is usually caused by a voltage regulator, adequate for a standard QL but which failed when peripherals were added and the machine was hot. My older QL, a genuine D11 issue, used to suffer this problem in the summer months.

Voltage regulators of this type are short-circuit proof in that,, if overloaded, their output "folds back": while not ceasing to work altogether, they no longer supply sufficient voltage to keep the QL going.

Also, working near their capacity, the regulators generate a large amount of heat and, if the heatsink cannot dissipate the heat fast enough, their rated power capacity is reduced dramatically.

Replacing the standard voltage regulator, rated at 1.5 to 2 amps, is a simple and inexpensive — £2 to £4 — cure for this problem. The new regulator, with a higher capacity, should be either an L78S05CY or MC78T05CT, in the same package type (TO-220).

A new voltage regulator can be fitted by removing the top of the QL, being careful of the wires connecting the top with the motherboard; removing the screw from the voltage regulator, mounted on the large black heatsink located directly behind the Microdrives and pulling the connector from the three-legged regulator. No soldering is required.

Possible culprit

Another possible culprit is that corruption has occurred during the transfer of the 2.3 programs from Microdrive to disc, making their running unreliable. The 2.3 programs should be re-transferred, preferably late in the evening when there is less chance of spikes on the mains.

It could also be that program failure is a combination of the two problems. As I have used Psion 2.3 programs from disc since their release and without difficulties, I would suggest that either or both of the two solutions is considered.

Another reader, apparently having difficulties getting his printer driver to work, indicates that he is relatively new to the QL and has recently added an MCS-QLD-5 Memodisk interface and Cumana disc drives to his system. Other comments indicate that there is some confusion as to what the three *_bas* programs do. Provided he is not using pre-version 2.0 Psion programs, he can ignore *CONVERT_bas* completely.

When Psion programs are transferred for use with disc-based systems, users have one of two options; either they can use the toolkits supplied with most interfaces and add *FLP_USE MDV* to their boot programs, in which case, transfer from Microdrive to disc will require no re-configuring, or they can run Psion *CONFIG_bas* to alter the drive calls.

Taking the latter course, when *CONFIG_bas* is run, you will be asked two initial questions, the name of the program to be re-configured and the drive on which it is located.

After looking at the designated program, *CONFIG_bas* will report the current default devices and ask for three drive IDs — the System Information, the device on which the printer driver is kept; the location of the HELP file; and the default drive for Data, where the finished document is to be stored on default and *def_tmp* files.

If a program is not re-configured and the user does not use drive emulation, when Quill tries to access a non-available HELP file it will report that it is not available and then continue.

If the *printer_dat* file is not available in its expected drive, a document will be printed in whichever font mode the printer happens to be when the command is entered and will ignore most of the control codes used in Quill, such as bold, subscript, *ad infinitum*.

Users who print large numbers of documents in a single sitting may care to consider configuring the Psion programs to call for their *printer_dat* drivers from a RAM disc — either included in most disc interfaces or available as a separate program — rather than conventional drives. Besides saving wear and tear, access is faster and it frees the drive for other uses. If this idea appeals

Sensational

QL SOFTWARE OFFERS

Reversi	Only	£4.52
Decision Maker	Only	£9.04
Macro Assembler	Only	£15.82
Gardener	Only	£9.04
Touch and Go	Only	£4.52
Home Finance	Only	£9.04
Meteor Storm	Only	£4.52
Monitor	Only	£9.04

All items are excluding V.A.T.
Orders under £20 + £2 Carriage

cpc

'Good people to deal with'

Dept. QL, 194-200 North Road, Preston, Lancashire. PR1 1YP

(0772) 555034

CARET COMPUTERS

BACK BY POPULAR DEMAND
JAN JONES: QL SuperBASIC, The Definite Handbook
*** Price £9.95 order your copy NOW. DEALER enquires WELCOME**
Recommended by the authors of Qliberator & as seen in QL WORLD

DISK 3.5" DS/DD	£10	for £18.50 or 20 for £35
M'DRIVES CART(new)	£10	for £17.50 or 20 for £30
MIRACLE TRUMP CARD	£199.00	(768K ram)
MIRACLE EXPANDERAM	£99.00	(512K ram)
MIRACLE MODEM	£49.00	
MIRACLE PRINTER INTERFACE	£18.50	
INTENAL MEMORY	£99.00	(512K) fitted £125
QL SERVICE MANUALS	£22.00	
INTERNAL CLOCK	£21.50	(e-z fitting no mess)
SCHON KEYBOARDS	£57.00	

SOFTWARE

TURBO (inc toolkit)	£90.00	SUPER MEDIA MANAGER	£35.00
THE EDITOR	£29.95		
(SPECIAL EDITION)	£39.95	EYE-Q	29.95
QLIBERATOR (v3.1)	£60.00	QL MACRO ASSEMBLER	£35.00
QL ED/ASM (ADDER)	£9.95	QL ED/ASM (COLIN OPPIE)	£29.95
QL ASSEMBLER (GST)	£15.00	CARTRIDGE DOCTOR (ADDER)	£9.95
QL MONITOR (Qjump)	£19.95	QL PROJECT PLANNER	£19.95
QL HOME FINANCE	£19.95	CASH TRADER	£55.00
PSION SUITE 2.3	£9.00	BASIC ALLY (TALENT)	£25.00
QDRAW	£12.95	CAD PACK	£9.95
JUNGLE EDDIE	£13.95	ZKUL	£12.95
3D SLIME	£12.95	LOST PHAROAH	£12.00
STRIP POKER	£13.95	MATCH POINT	£12.95
QL REVERSI	£9.95	QL CLASSIC ADVENTURES	£9.95
QL CAVERNS	£9.95	QL BRIDGE	£9.95

BOOKS

QL ADVANCE USER GUIDE (ADDER)	£ 14.95
QL ASSEMBLY PROGRAMMING (COLIN OPPIE)	£19.95
ASSORT HUTCH BOOKS	£1.75

SPECIAL OFFERS
a) BUY 'OMEGA the Game' disc ver11 for £15.95 and RECEIVE 2 free disk. (offer only on 3.5' disc)
b) 'OMEGA the Game' microdrive VER11 now only £14.95 (dealer enquires welcome)
c) 3.5' disc labels 100 for £7.00
d) Adder EDITOR/ASM £7.95. not suitable for expanded memory

all products subject to availability

Please find My Enclosed Cheque for .....

Send me .....

My Name .....

Address .....

.....

CARET COMPUTERS 22 CAVENDISH RD. LONDON NW6 7XP
Tel Daytime 01-267-8610 (ask for Mike Weaver)
Evenings 01-459 2267(19.30 till 21.30)
PRESTEL MBX 014592267 VISA/ACCESS CHEQUES POSTAL ORDERS

Access VISA

BYTEBACK

THE QL SPECIALIST

GAMES		ONLY
Type-22	Talent	16.00
Deathstrike	Talent	12.50
Hover Zone	Talent	12.50
Farmer	Talent	12.50
QL Defusion	Realtime	8.50
Stone Raider II	Microdeal	12.50
Steve Davis Snooker	CDS	12.50
Bridge Player II	C.P.Soft	16.00
3D Slime	Datalink	11.00
Blocklands	Digital	8.50
Droidzone	Digital	8.50
Super Backgammon	Digital	11.00
Spook	Eidersoft	9.00
Karate	Eidersoft	12.50
BJ in 3D Land	Eidersoft	7.00
BJ The Return	Eidersoft	8.00
Citadel	Eidersoft	7.00
Zapper and Eagle	Eidersoft	7.00
Hyperdrive	English	12.50
Scrabble	L. Genius	12.50
Flight simulator	Microdeal	17.00
Lands of Havoc	Microdeal	17.00
3D Chess	Psion	17.00
Match Point	Psion	12.50
Super Croupier	Pyramide	12.50
Mister Smith	Pyramide	9.50
Vroom	Pyramide	13.00
Wanderer	Pyramide	17.00
Othello 3D	Pyramide	17.00
Knight Flight	Realtime	12.50
Dragonhold	Rubicon	12.50
Tank Busters	Sellasoft	14.00
Quboids	Sinclair	8.00
Bounder	Sinclair	8.00
Meter Storm	Sinclair	8.00
Strip Poker	Talent	12.50
Oxford Trivia	Talent	15.00
Jungle Eddi	Talent	12.50
Lost Pharoah	Talent	12.50

PROGRAMMING		ONLY
Supercharge	Digital	42.00
Turbo(+ Toolkit)	Digital	84.00
Better Basic	Digital	21.00
Superforth	Digital	34.00
Assembler Dev.Kit	Metacomco	34.00
Lisp Dev.Kit	Metacomco	49.00
C Dev.Kit	Metacomco	79.00
BCPL Dev.Kit	Metacomco	49.00
Pascal Dev.Kit	Metacomco	75.00
Basic-Ally	Talent	25.00
Assembler Workbench	Talent	21.00

ART AND DESIGN		ONLY
Datacad QL	Datakey	9.50
Eye-Q	Digital	25.00
Super Srite Gen	Digital	25.00
Sign Designer	D.S. Ent's	16.00
The Designer	Pyramide	17.00
Graphic Toolkit	Pyramide	12.50
Nucleon	Pyramide	17.00
QL Peintre	Pyramide	12.50
Viewpoint 3D	Rubicon	17.00
PCB Designer 1	Talent	85.00
PCB Designer 2	Talent	45.00
Graphical Plus	Talent	21.00
TechniQL	Talent	43.00
Technikit	Talent	21.00

ADVENTURES		ONLY
Horrorday	Talent	12.50
Quill System	Gilsoft	16.00
Executive Adv.	Gemini	12.50
Aquanaut 471	Microdeal	17.00
Mortville Manor	Pyramide	17.00
Nemesis	Talent	12.50
Zkul	Talent	12.50
West	Talent	12.50

UTILITIES AND BUSINESS		ONLY
Front Page	Gap	21.00
Task Master	Sector	25.00
Spellbound	Sector	30.00
Desktop Publisher	Digital	59.00
Wimp Designer	Talent	13.00
Super Astrologer	Digital	21.00
Pro Astrologer	Digital	45.00
Super Media Mgr	Digital	34.00
Cart. Media Mgr	Digital	21.00
The Editor	Digital	25.00
Editor Special Ed.	Digital	35.00
ICE	Eidersoft	25.00
Icicle	Eidersoft	12.50
ICE Toolkit	Eidersoft	11.00
Choice	Eidersoft	14.50
Artice	Eidersoft	12.50
Qspell	Eidersoft	19.00
Qspell (disk)	Eidersoft	23.00
Bag of Tricks 1&2	Pyramide	10.00
Touch Typist	Sector	11.00
Qimp	Talent	21.00
Sideways	Talent	17.00
QL Toolset	Talent	17.00
Cartridge Doctor	Talent	12.50
Cosmos	Talent	12.50
Tasprint	Tasman	18.00

HARDWARE	ONLY		ONLY
Joystick adaptor	4.50	Microdrive Cartridges	1.80
The ARCADE joystick	17.00	4 Cartridges (in wallet)	7.50
Quickshot II joystick	8.00	20 Cartridges in Transform Box	39.00
Quickshot TURBO joystick	13.00	Transform storage box	4.50
Mouse + ICE + ARTice	79.00	Miracle Expanderam 512	89.00
Miracle Trump card	189.00	Miracle Centronics Interface	18.00
Miracle QL Modem	45.00	Schon QL Keyboard	55.00
Star NL10 Printer	229.00	Atari 520 ST-FM	359.00

BARGAIN BASEMENT	ONLY
4 Game Pack (BJ in 3D Land + Citadel + Zapper + Eagle)	14.00
2 Game Pack (Bounder + Quboids) ★ Limited Stocks ★	13.00
ICE eprom + Artice + Toolkit + Choice + Icicle	72.00
ICE eprom + Artice + Toolkit + Choice + Icicle	72.00

Best prices on QL EXPORT ORDERS to EUROPE and the WORLD! Write for details.

ALL PRICES INCLUDE VAT AND FIRST CLASS POST
TELEPHONE (0636) 79097 FOR PERSONAL ASSISTANCE
Send Cheque/PO to:

BYTEBACK **6, Mumby Close, Newark, Notts NG24 1JE**

·PSION· SOLUTIONS·

to you, add the following to the Psion boot programs:

1 FORMAT RAM1_2: REMark Uses a tiny bit of memory
2 COPY drive*_printer_dat TO RAM1_printer_dat*

The first report of the current Psion Solutions series, detailing a method for having all the Psion programs on a single medium, seems to have sparked considerable interest, reflecting varying degrees of understanding and confusion as to what was accomplished by the method outlined.

First, there is nothing, except common sense, preventing anyone putting any number of programs on a single volume and calling the various files "A", "B", "C" and so on. File-naming restrictions arise when programs use external routines specified from within their program structure which are not readily-accessible. Such is the case with, among many other programs, the Psion suite and two such files are *printer_dat* and *def_tmp*.

File access

The first of these, common to three of the Psion programs, is a requirement for a file called *printer_dat*, accessed when the PRINT command is given. This file contains the codes applicable to an individual printer and is generally configured by the user for individual applications.

The second, a Quill requirement, is for an external file called *def_tmp*. With 128K of RAM this file is generated automatically when the content of a current document reaches approximately 1,200 words.

In a manner of speaking the QL memory is looped externally through the *def_tmp* file so that a document of more than about 1,200 words is never entirely contained in memory. With a 640K machine the upper limit is about 41,000 words.

Multi-tasking with more than one Quill can cause considerable problems if more than one Quill requires a *def_tmp* file on the same drive. Looking more closely at the solution provided in the May issue of *QL World*, if you use the default printer driver as supplied for Quill, Abacus and Archive, having only one *printer_dat* file is acceptable.

The standard *printer_dat* file makes three assumptions. First, that you are using a printer which is compatible with one of the those listed in the *Install_bas* routine; second, you use the pica-sized — 10 characters per inch — font for all a your printing requirements and that all the translate codes are common to the three Psion programs.

If any of those requirements is not satisfactory, another means must be found to make each of the printer drivers program-specific. My requirements demanded individual drivers, each with exceptional file names; these drivers are built to satisfy the codes used by my Canon PW1080A printer which, incidentally, has the DIP switches set to the American character set on default. The names for each driver were altered in the respective programs, using the DP *Editor*.

Parameter	Quill	Abacus	Archive
Driver name	*QUprint*	*ABprint*	*ARprint*
Preamble — set fonts	*Elite*	*Condensed*	*Pica*
Postamble	*None*	*Back to Elite*	*Back to Elite*
Bold on	*Pica bold*	—	—
Bold off	*Elite*	—	—
Underline on	*Standard*	—	—
Underline off	*Standard*	—	—
Subscript on	*Italics*	—	—
Subscript off	*Elite*	—	—
Superscript on	*Condensed*	—	—
Superscript off	*Elite*	—	—
Trans 1	*£ sign*	*£ sign*	*£ sign*
Trans 2	*Doublestrike*	*Pica bold on*	*Bold proportional*
Trans 3	*Elite*	*Condensed*	*Bold off*
Trans 4	*Enlarged On*	*Bold on*	*Bold, condensed*
Trans 5	*Degree symbol*	*Subscript on*	*Std Pica*
Trans 6	*Doub Strk off*	*Subscript off*	*Underline on*
Trans 7	—	*Degree sign*	*Underline off*
Trans 8	—	—	*Enlarged on*
Trans 9	—	—	*Enlarged off*
Trans 10	—	—	*Degree sign*

Each of the foregoing, used in various combinations, gives me access to the entire range of Canon fonts which satisfy most of my printing requirements. I multi-task with two slightly different versions of Quill. One defaults to flp2_ for its Data drive, the other, to RAM1_. If the latter does not produce a *def_tmp* file, I change the drive designator when I save or load a file.

THE P+R:O=G<S

If you have a program that is worthy of consideration, send it to 'The Progs', Sinclair QL World, Greencoat House, Francis Street, London SW1P 1DG. We pay for everything published at the usual page rates — £80 per thousand words.

Program of the month

QL Roulette by Santiago Rubio

This Las Vegas Roulette simulation allows you to try your hand at casino gambling without losing your shirt. Though written in Spanish, pressing '2' at the start of the program will print all the information in English.

The bets are set using the cursor keys and pressing ENTER once you have made your choice. The process can be repeated up to six times, following the instructions in the screen windows.

Nine betting lots are available, plus six more relating to single bets such as Black, Red, Even, Odd, Lack and Pass. All have different limits and payouts. When 'O' appears the bank returns half of the stacks placed to single lots.

A special feature of the program is called the Leigh Pattern and is based on a description of how to beat the bank in Norman Leigh's novel *Thirteen Against The Bank*. Six people around a gambling table bet simultaneously to the six single lots — Black, Red, Even, Odd, Lack, Pass. As they do so, they write the following series in their notebooks — one per person — and bet the sum of the higher or lower figures:

B	*R*	*E*	*D*	*L*	*P*
—	—	—	—	—	—
1	*1*	*1*	*1*	*1*	*1*
2	*2*	*2*	*2*	*2*	*2*
3	*3*	*3*	*3*	*3*	*3*
4	*4*	*4*	*4*	*4*	*4*

In the first throw, the bets are obviously the same for all players — ie., 1+4 = 5. If the roulette wheel gave, say, 14 — Red, Even, Lack — as the result, the winners would add their winnings to the list and the losers would cross out the figures lost, so that the notebooks would read:

B	R	E	D	L	P
—	—	—	—	—	—
/	1	1	/	1	/
2	2	2	2	2	2
3	3	3	3	3	3
/	4	4	/	4	/
	5	5	5		

The bets for the second throw are now 5, 6, 6, 5, 6, 5 units respectively. If the result was, say, 26 — Black, Even, Pass — the notebooks would show:

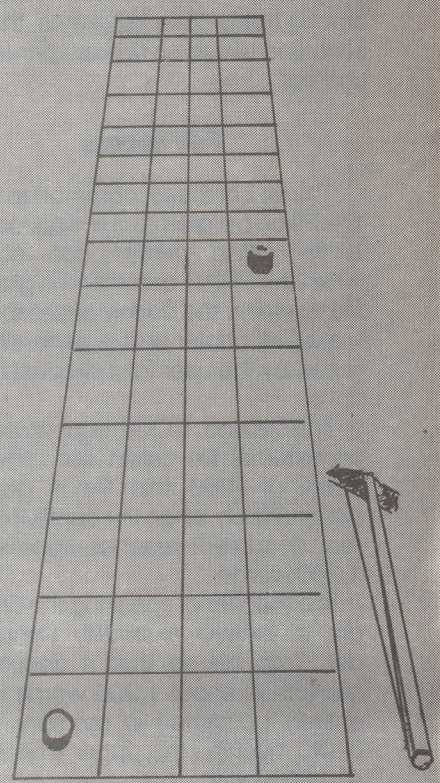
B	R	E	O	L	P
—	—	—	—	—	—
/	/	1	/1	/	/
2	2	2	/2	2	2
3	3	3	/3	3	3
/	4	4	/4	4	/
5	/	5		/	5
		6			

Anybody who crosses out all the figures, such as Odd, must start a new series of 1, 2, 3, 4.

So the procedure is repeated. If at any time only one figure remains in the list, it is the amount of the next bet. When one particular stack goes over the limit — 360 units for single lots — the gambler must start again with a new 1, 2, 3, 4 series. The game will finish when you either run out of money or break the bank.

Put your QL to work. The program features the developed Leigh Pattern, so you can watch the amazing results. Press <F1> for fair play or <F2> for the Leigh Pattern mode. You can move from one mode to another at the end of each throw.

The <F3> key allows you to send results to the printer, which is turned off with <F4>. If you want to analyse the progress of the game, <F5> will give statistics, which can be printed with <F3>. Finally, <SPACE> is used to force the roulette wheel to spin.



```
100 REMark ******************************************
110 REMark ****                QL                ****
120 REMark ****              RULETA              ****
130 REMark ****        Por Santiago Rubio        ****
140 REMark ****              1987                ****
150 REMark ****           Version 1.2            ****
160 REMark ******************************************
170 :
```

```
180 WINDOW 512,256,0,0:MODE 4:PAPER 4:INK 0:CLS
190 CLEAR:RANDOMISE:CSIZE 0,0
200 DIM cla$(27,9,2),clase$(27,9,9),vlr%(27,9,4)
210 DIM apt$(6,9),aps$(6,9),grp%(6,4),pos%(27,9)
220 DIM cnt(6),gan(6),may(6),men(6),nrueda%(36)
230 DIM val(50,6),ord(6),col%(6),fil%(6),nro(36)
240 fdon=0:banc=0:cntt=0:gant=0:impr=0:pret=0
250 njug=0:a=1:roj=0:par=0:pas=0:ett=0:pct=0
260 do1=0:do2=0:do3=0:co1=0:co2=0:co3=0
270 str$='ZPLCBTRCUZZSECLDODCDDFLPSPRIMRJNG'
280 :
290 abrir_canales
300 AT 12,25
310 PRINT 'Textos en castellano, presionar <1>'
320 AT 14,25
330 PRINT '     English texts,  press <2>
340 REPeat idioma
350 txt=CODE(INKEY$(-1))
360 SELect ON txt
370 =49
380 PRINT#6;'          Presionada tecla  1'
390 texto_esp:EXIT idioma
400 =50
410 PRINT#6;'            Key 2  selected'
420 texto_ing:EXIT idioma
430 =REMAINDER :aviso
440 PRINT#6;'  Eleccion no valida/Invalid choice'
450 END SELect
460 END REPeat idioma
470 FOR f=12 TO 14:AT f,0:PRINT FILL$(' ',85)
480 PRINT#8;ta$:PRINT#9;tb$
490 informe
500 PRINT#6;t2$
510 leer_matrices
520 PRINT#6;t3$:aviso:PAUSE
530 FOR f=4 TO 22:AT f,0:PRINT FILL$(' ',85)
540 dibujar_pantalla
550 texto_3:texto_4b:texto_5
560 cantidad_jugador
570 cantidad_banca
580 iniciar_juego
590 :
600 DEFine PROCedure iniciar_juego
610 CLS#6:PRINT#6;t21$
620 actualizar_fondos
630 elegir_opciones
640 END DEFine
650 :
660 DEFine PROCedure apuestas_libres
670 CLS#3
680 REPeat jugar
690 REPeat colocar
700 PRINT#6;t5$:PRINT#9;t1$
710 mover_cursor
720 escribir_apuesta
730 escribir_cantidad
740 PRINT#6;t3$:PRINT#9;t12$
750 IF INKEY$(-1)=' ':EXIT colocar
760 IF cntt=fdon:PRINT#9;t16$:EXIT colocar
770 CLS#3:a=a+1:IF a>6:a=6:EXIT colocar
780 END REPeat colocar
790 elegir_numero
800 PRINT#9;t10$
810 verificar_apuestas
820 actualizar_fondos
830 PRINT#6;t3$
840 cambiar 0
850 IF ett=1:texto_4
860 IF fdon<=0:EXIT jugar
870 IF banc<=0:EXIT jugar
880 borrar_variables
890 END REPeat jugar
900 reiniciar
910 END DEFine
920 :
930 DEFine PROCedure abrir_canales
```

```
940 BLOCK 444,20,36,235,0
950 BLOCK 444,30,36,5,0
960 OPEN #6;scr_444x20a32x231:CSIZE#6;2,1
970 OPEN #7;scr_444x30a32x1
980 OPEN #8;scr_444x20a32x1  :CSIZE#8;2,1
990 OPEN #9;scr_444x10a32x21 :CSIZE#9;2,0
1000 OPEN#10;scr_32x20a71x90  :CSIZE#10;3,1
1010 OPEN#11;con_48x10a287x156:CSIZE#11;0,0
1020 OPEN#12;con_48x10a287x186:CSIZE#11;0,0
1030 FOR ch1=6 TO 9:PAPER#ch1;2:INK#ch1;4:CLS#ch1
1040 END DEFine
1050 :
1060 DEFine PROCedure leer_matrices
1070 RESTORE 1090
1080 FOR m=1 TO 36:READ nrueda%(m)
1090 DATA 32,15,19,4,21,2,25,17,34,6,27,13
1100 DATA 36,11,30,8,23,10,5,24,16,33,1,20
1110 DATA 14,31,9,22,18,29,7,28,12,35,3,26
1120 FOR x=1 TO 27
1130 FOR y=1 TO 9
1140 cla$(x,y)=' ':vlr%(x,y,1)=99
1150 END FOR y
1160 END FOR x
1170 FOR y=1 TO 5
1180 cla$(1,y)='PL':clase$(1,y)=t50$:vlr%(1,y,1)=0
1190 END FOR y
1200 RESTORE 1250
1210 FOR y=1 TO 6
1220 READ cla$(2,y),clase$(2,y)
1230 FOR w=1 TO 4:READ vlr%(2,y,w)
1240 END FOR y
1250 DATA 'CB',t51$,0, 1,99,99
1260 DATA 'TR',t52$,0, 1, 2,99
1270 DATA 'CB',t51$,0, 2,99,99
1280 DATA 'TR',t52$,0, 2, 3,99
1290 DATA 'CB',t51$,0, 3,99,99
1300 DATA 'CU',t53$,0, 1, 2, 3
1310 FOR x=3 TO 25 STEP 2
1320 FOR y=1 TO  5 STEP 2
1330 cla$(x,y)='PL':clase$(x,y)=t50$
1340 vlr%(x,y,1)=(3*x+y-8)/2
1350 END FOR y
1360 END FOR x
1370 FOR x=3 TO 25 STEP 2
1380 FOR y=2,4
1390 cla$(x,y)='CB':clase$(x,y)=t51$
1400 vlr%(x,y,1)=(3*x+y-9)/2
1410 vlr%(x,y,2)=(3*x+y-7)/2
1420 END FOR y
1430 END FOR x
1440 FOR x=4 TO 24 STEP 2
1450 FOR y=1,3,5
1460 cla$(x,y)='CB':clase$(x,y)=t51$
1470 vlr%(x,y,1)=(3*x+y-11)/2
1480 vlr%(x,y,2)=(3*x+y- 5)/2
1490 END FOR y
1500 END FOR x
1510 FOR x=3 TO 25 STEP 2
1520 cla$(x,6)='TR':clase$(x,6)=t52$
1530 vlr%(x,6,1)=(3*x-7)/2
1540 vlr%(x,6,2)=(3*x-5)/2
1550 vlr%(x,6,3)=(3*x-3)/2
1560 END FOR x
1570 FOR x=4 TO 24 STEP 2
1580 FOR y=2,4
1590 cla$(x,y)='CU':clase$(x,y)=t53$
1600 vlr%(x,y,1)=(3*x+y-12)/2
1610 vlr%(x,y,2)=(3*x+y-10)/2
1620 vlr%(x,y,3)=(3*x+y- 6)/2
1630 vlr%(x,y,4)=(3*x+y- 4)/2
1640 END FOR y
1650 END FOR x
1660 FOR x=4 TO 24 STEP 2
1670 cla$(x,6)='SE':clase$(x,6)=t54$
1680 vlr%(x,6,1)=(3*x-10)/2
1690 END FOR x
```

```
1700 FOR y=1,3,5
1710 cla$(27,y)='CL':clase$(27,y)=t55$
1720 vlr%(27,y,1)=(y+1)/2
1730 END FOR y
1740 FOR y=2,4
1750 cla$(27,y)='DC':clase$(27,y)=t56$
1760 vlr%(27,y,1)=y/2
1770 END FOR y
1780 FOR u=3,11,19
1790 FOR x=u TO u+6
1800 cla$(x,7)='DO':clase$(x,7)=t57$
1810 vlr%(x,7,1)=(3*u-7)/2
1820 END FOR x
1830 END FOR u
1840 FOR x=10,18
1850 cla$(x,7)='DD':clase$(x,7)=t58$
1860 vlr%(x,7,1)=(3*x-28)/2
1870 END FOR x
1880 FOR u=3,4,5
1890 RESTORE 1940
1900 FOR x=u TO u+20 STEP 4
1910 READ cla$(x,9),clase$(x,9),vlr%(x,9,1)
1920 END FOR x
1930 END FOR u
1940 DATA 'FL',t59$, 1,'PR',t60$, 2,'RJ',t61$, 1
1950 DATA 'NG',t62$, 2,'IM',t63$, 1,'PS',t64$,19
1960 END DEFine
1970 :
1980 DEFine PROCedure dibujar_pantalla
1990 LOCal l,s,t,nu,u,v,w,z
2000 CLS#6
2010 oxc=25:oyc=61:ra=25:oxn=81:oyn=95:ju=0
2020 CIRCLE oxc,oyc,ra:CIRCLE oxc,oyc,ra-5
2030 FILL 1:CIRCLE oxc,oyc,ra-10:FILL 0
2040 FOR ang=1.5*PI TO 3.5*PI STEP 2*PI/37
2050 yc=ra*SIN(ang):xc=ra*SIN(PI/2-ang)
2060 LINE oxc,oyc TO oxc+xc,oyc+yc
2070 END FOR ang
2080 FOR ang=1.5*PI TO (255/74)*PI STEP 2*PI/37
2090 xn=78*SIN(PI/2-ang):yn=58*SIN(ang)
2100 CURSOR oxn+xn,oyn+yn
2110 IF ju=0:INK 6
2120 IF ju>0:INK (ju MOD 2)*2
2130 PRINT nrueda%(ju)
2140 ju=ju+1
2150 END FOR ang
2160 INK 7:CIRCLE oxc,oyc,ra-18
2170 FOR l= 38 TO  98 STEP 60:BLOCK 280,1,210,1,6
2180 FOR l= 58 TO  78 STEP 20:BLOCK 260,1,230,1,6
2190 FOR l=118 TO 138 STEP 20:BLOCK 240,1,230,1,6
2200 FOR l=210 TO 490 STEP 20:BLOCK 1,60,1,38,6
2210 FOR l=230 TO 470 STEP 80:BLOCK 1,40,1,98,6
2220 FOR l=270 TO 430 STEP 80:BLOCK 1,20,1,118,6
2230 FOR s=231 TO 451 STEP 20
2240 IF s=231 OR s=351:RESTORE 2430
2250 FOR t= 39 TO  79 STEP 20
2260 READ color
2270 BLOCK 19,19,s,t,color
2280 END FOR t
2290 END FOR s
2300 nu=1
2310 OVER-1
2320 CURSOR 217,64
2330 INK 6:PRINT 0
2340 FOR u=237,257,277,294 TO 454 STEP 20,477
2350 IF u=477:nu=1
2360 FOR v=44 TO 84 STEP 20
2370 CURSOR u,v
2380 PRINT nu
2390 nu=nu+1
2400 END FOR v
2410 END FOR u
2420 OVER 0
2430 DATA 2,0,2,0,2,0,2,0,2,0,0,2,0,2,0,2,0,2
2440 FOR w=243,323,403
2450 CURSOR w,103
2460 INK 0:PRINT;t57$;(w-163)/80
2470 END FOR w
2480 RESTORE 2540
2490 FOR z=235,280,318,355,395,438
2500 READ sen$
2510 CURSOR z,123
2520 PRINT sen$
2530 END FOR z
2540 DATA t59$,t60$,t61$,t62$,t63$,t64$
2550 BLOCK 135,60,9,169,0
2560 BLOCK 210,80,153,150,0
2570 BLOCK 135,60,372,169,0
2580 OPEN #3;con_135x60a5x165  :CSIZE#3;0,0
2590 OPEN #4;con_210x80a149x146:CSIZE#4;0,0
2600 OPEN #5;scr_135x60a368x165:CSIZE#5;0,0
2610 FOR ch1=3 TO 5,11,12
2620 PAPER#ch1;2:INK#ch1;4:CLS#ch1
2630 END FOR ch1
2640 END DEFine
2650 :
2660 DEFine PROCedure cantidad_jugador
2670 PRINT#6;t11$
2680 intr 1
2690 fdon=fdon+ctd$
2700 END DEFine
2710 :
2720 DEFine PROCedure cantidad_banca
2730 PRINT#6;t18$
2740 intr 10
2750 banc=banc+ctd$
2760 END DEFine
2770 :
2780 DEFine PROCedure intr (r)
2790 REPeat ciclo
2800 CLS#(r+98)/9
2810 INPUT#(r+98)/9!!ctd$
2820 IF ctd$='':aviso
2830 comprobar_entrada
2840 IF NOT falta:EXIT ciclo
2850 END REPeat ciclo
2860 IF ctd$<r*1000 OR ctd$>r*10000:aviso:intr r
2870 END DEFine
2880 :
2890 DEFine PROCedure informe
2900 AT 4,0
2910 PRINT i0$,,k0$\i1$,,k1$\i2$,,k2$
2920 PRINT i3$,,k3$\i4$,,k4$\i5$,,k5$
2930 PRINT i6$,,k6$\i7$,,k7$\i8$,,k8$
2940 PRINT i9$,,k9$\j0$,,10$\j1$,,11$
2950 PRINT j2$,,12$\j3$,,13$\j4$,,14$
2960 PRINT j5$,,15$\j6$,,16$\j7$,,17$
2970 END DEFine
2980 :
2990 DEFine PROCedure texto_3
3000 CLS#3:PRINT#3;t36$\t37$\t38$\t39$\t40$\t41$
3010 END DEFine
3020 :
3030 DEFine PROCedure texto_4
3040 CLS#4:PRINT#4;t30$\t31$
3050 END DEFine
3060 :
3070 DEFine PROCedure texto_4b
3080 CLS#4:PRINT#4;\t26$\t27$\\t28$\t29$
3090 END DEFine
3100 :
3110 DEFine PROCedure texto_5
3120 CLS#5:PRINT#5;t42$\t43$\t44$\t45$\t46$\t47$
3130 END DEFine
3140 :
3150 DEFine PROCedure elegir_opciones
3160 REPeat opcion
3170 opc=CODE(INKEY$(-1))
3180 SELect ON opc
3190 =232:CLS#8:CLS#9:texto_4:apuestas_libres
3200 =236:CLS#8:CLS#9:texto_4:sistema_Leigh
3210 =240:impr=1:PRINT#6;t20$:PAUSE 99:PRINT#6;t4$
3220 =244:impr=0:PRINT#6;t24$:PAUSE 99:PRINT#6;t4$
3230 =248:estadisticas
3240 =REMAINDER :aviso
3250 END SELect
3260 END REPeat opcion
3270 END DEFine
3280 :
3290 DEFine PROCedure mover_cursor
3300 s=215:t=123:ficha
3310 REPeat siempre
3320 REPeat mover
3330 iky=CODE(INKEY$(-1))
3340 ficha
3350 SELect ON iky
3360 =192:s=s-10:IF s<215:s=475
3370 =208:t=t-10:IF t<43 :t=123
3380 =200:s=s+10:IF s>475:s=215
3390 =216:t=t+10:IF t>123:t=43
3400 = 10:ficha:EXIT mover
3410 =REMAINDER :aviso
3420 END SELect
3430 ficha
3440 END REPeat mover
3450 x=(s-205)/10:y=(t-33)/10
3460 col%(a)=s:fil%(a)=t
3470 apt$(a)=cla$(x,y)
3480 aps$(a)=clase$(x,y)
3490 FOR w=1 TO 4:grp%(a,w)=vlr%(x,y,w)
3500 IF grp%(a,1)<>99:EXIT siempre
3510 aviso
3520 END REPeat siempre
3530 PRINT#6;t6$
```

```
3540 END DEFine
3550 :
3560 DEFine PROCedure escribir_apuesta
3570 AT#4;a+1, 1:PRINT#4;aps$(a)
3580 menor=grp%(a,1)
3590 str=apt$(a) INSTR(str$)
3600 SELect ON str
3610 = 2 TO 8 :mayor=grp%(a,str/2)
3620 =12,24   :mayor=menor+str-7
3630 =16,22   :mayor=menor+str-5
3640 =20,30   :mayor=menor+6*str/5-1
3650 =14,32   :mayor=menor+33
3660 =18,26,28:mayor=menor+34
3670 END SELect
3680 may(a)=mayor:men(a)=menor
3690 AT#4;a+1,13-LEN(men(a))
3700 PRINT#4;men(a);'-';may(a)
3710 END DEFine
3720 :
3730 DEFine PROCedure escribir_cantidad
3740 REPeat ciclo
3750 AT#4;a+1,17:PRINT#4;FILL$(' ',17)
3760 AT#4;a+1,17:INPUT#4;ctd$
3770 comprobar_entrada
3780 IF NOT falta
3790 comprobar_maximo
3800 IF ctd$>max
3810 PRINT#6;t7$;max;t9$
3820 ELSE
3830 comprobar_fondos
3840 IF hay_fondos=1:EXIT ciclo
3850 END IF
3860 END IF
3870 aviso
3880 END REPeat ciclo
3890 AT#4;a+1,17:PRINT#4;FILL$(' ',17)
3900 AT#4;a+1,23-LEN(cnt(a)):PRINT#4;cnt(a)
3910 END DEFine
3920 :
3930 DEFine PROCedure comprobar_entrada
3940 falta=1:fa=0
3950 FOR j=1 TO LEN(ctd$)
3960 IF ctd$(j)>='0' && ctd$(j)<='9':fa=fa+1
3970 IF fa=j:falta=0:ELSE falta=1:aviso
3980 END FOR j
3990 END DEFine
4000 :
4010 DEFine PROCedure comprobar_maximo
4020 str=apt$(a) INSTR(str$)
4030 SELect ON str
4040 = 2 TO 12:max=str*10
4050 =14 TO 16:max=240
4060 =18 TO 20:max=480
4070 =22 TO 32:max=360
4080 END SELect
4090 END DEFine
4100 :
4110 DEFine PROCedure comprobar_fondos
4120 hay_fondos=0
4130 cnt(a)=ctd$
4140 cntt=cntt+cnt(a)
4150 IF cntt<=fdon:hay_fondos=1
4160 IF cntt>fdon
4170 cntt=cntt-cnt(a)
4180 PRINT#6;t8$;fdon-cntt;t9$
4190 END IF
4200 END DEFine
4210 :
4220 DEFine PROCedure elegir_numero
4230 PRINT#6;t17$
4240 BEEP 0,25,255,10,150,100
4250 FOR nr=0 TO 36,0 TO 36, 0 TO 36,RND(36)
4260 numero=nrueda%(nr)
4270 PRINT#10;numero
4280 END FOR nr
4290 PAN#8;36
4300 CLS#6:BEEP
4310 IF numero<10
4320 CLS#10
4330 CURSOR#10,9,0
4340 PRINT#10;numero
4350 AT#8;0,0:PRINT#8;CHR$(188)!numero
4360 ELSE
4370 PRINT#10;numero
4380 AT#8;0,0:PRINT#8;CHR$(188);numero
4390 END IF
4400 resultado
4410 njug=njug+1
4420 nro(numero)=nro(numero)+1
4430 END DEFine
4440 :
4450 DEFine PROCedure verificar_apuestas
4460 FOR c=1 TO a
4470 IF apt$(c)='PL':apt_mul 36
4480 IF apt$(c)='CB':apt_mul 18
4490 IF apt$(c)='TR':apt_mul 12
4500 IF apt$(c)='CU':apt_mul  9
4510 IF apt$(c)='SE':apt_sen  0, 5,1,6
4520 IF apt$(c)='CL':apt_sen  0,33,3,3
4530 IF apt$(c)='DC':apt_sen  1,34,3,1.5
4540 IF apt$(c)='DC':apt_sen  0,33,3,1.5
4550 IF apt$(c)='DO':apt_sen  0,11,1,3
4560 IF apt$(c)='DD':apt_sen  0,23,1,1.5
4570 IF apt$(c)='FL':apt_sen  0,17,1,2
4580 IF apt$(c)='PS':apt_sen  0,17,1,2
4590 IF apt$(c)='PR':apt_sen  0,34,2,2
4600 IF apt$(c)='IM':apt_sen  0,34,2,2
4610 IF apt$(c)='RJ':apt_clr  1,35,2,2
4620 IF apt$(c)='NG':apt_clr  2,36,2,2
4630 gant=gant+gan(c)
4640 AT#4;c+1,33-LEN(gan(c)):PRINT#4;gan(c)
4650 END FOR c
4660 END DEFine
4670 :
4680 DEFine PROCedure actualizar_fondos
4690 fdon=fdon-cntt+gant
4700 banc=banc+cntt-gant
4710 FOR s=1 TO 5
4720 AT#5;s,15:PRINT#5;FILL$(' ',7)
4730 PAUSE 5
4740 END FOR s
4750 AT#5,1,21-LEN(cntt):PRINT#5;cntt
4760 AT#5,2,21-LEN(gant):PRINT#5;gant
4770 AT#5,3,21-LEN(fdon):PRINT#5;fdon
4780 AT#5,4,21-LEN(banc):PRINT#5;banc
4790 AT#5,5,21-LEN(njug):PRINT#5;njug
4800 IF impr=1:impr_juga
4810 END DEFine
4820 :
4830 DEFine PROCedure apt_mul(fctr)
4840 FOR w=1 TO 4
4850 IF numero=grp%(c,w):gan(c)=cnt(c)*fctr
4860 END FOR w
4870 END DEFine
4880 :
4890 DEFine PROCedure apt_sen(a0,a1,a2,fctr)
4900 IF numero=0
4910 IF fctr=2:gan(c)=INT(cnt(c)/2):RETurn
4920 END IF
4930 FOR f=a0 TO a1 STEP a2
4940 IF numero=grp%(c,1)+f:gan(c)=INT(cnt(c)*fctr)
4950 END FOR f
4960 END DEFine
4970 :
4980 DEFine PROCedure apt_clr(b0,b1,b2,fctr)
4990 IF numero=0:gan(c)=INT(cnt(c)/2):RETurn
5000 FOR f=b0 TO b1 STEP b2
5010 IF numero=nrueda%(f):gan(c)=cnt(c)*fctr
5020 END FOR f
5030 END DEFine
5040 :
5050 DEFine PROCedure borrar_variables
5060 FOR z=1 TO a
5070 s=col%(z):t=fil%(z):ficha
5080 apt$(z)=0:grp%(z,1)=0
5090 col%(z)=0:fil%(z)=0
5100 cnt(z)=0 :gan(z)=0
5110 may(z)=0 :men(z)=0
5120 AT#4;z+1,1:PRINT#4;FILL$(' ',34)
5130 END FOR z
5140 CLS#10
5150 FOR r=11 TO 13:AT r,29:PRINT FILL$(' ',6)
5160 cntt=0:gant=0:a=1:ett=0
5170 END DEFine
5180 :
5190 DEFine PROCedure ficha
5200 OVER-1
5210 BLOCK 10,10,s,t,2
5220 OVER 0
5230 END DEFine
5240 :
5250 DEFine PROCedure sistema_Leigh
5260 FOR a=1 TO 6:empezar_serie a
5270 PRINT#6;t19$:PRINT#9;t10$
5280 REPeat proceso
5290 FOR a=1 TO 6
5300 s=40*a+205:t=123
5310 x=4*a:y=9
5320 col%(a)=s:fil%(a)=t
5330 apt$(a)=cla$(x,y)
5340 aps$(a)=clase$(x,y)
5350 grp%(a,1)=vlr%(x,y,1)
5360 ficha
```

```
5370 IF ord(a)<1:empezar_serie a
5380 IF ord(a)=1:ctd$=val(1,a)
5390 IF ord(a)>1:ctd$=val(1,a)+val(ord(a),a)
5400 IF ctd$>360
5410 empezar_serie a
5420 ctd$=val(1,a)+val(ord(a),a)
5430 END IF
5440 escribir_apuesta
5450 comprobar_fondos
5460 IF NOT hay_fondos:EXIT proceso
5470 AT#4;a+1,23-LEN(cnt(a)):PRINT#4;cnt(a)
6080 READ res
6090 END FOR rs
6100 co$=t62$:pr$=t63$:mt$=t59$
6110 IF res =2         :co$=t61$:roj=roj+1
6120 IF numero MOD 2=0:pr$=t60$:par=par+1
6130 IF numero >=19    :mt$=t64$:pas=pas+1
6140 SELect ON numero
6150 = 1 TO 12:do1=do1+1
6160 =13 TO 24:do2=do2+1
6170 =25 TO 36:do3=do3+1
6180 END SELect
6190 IF numero MOD 3=1:co1=co1+1
6200 IF numero MOD 3=2:co2=co2+1
6210 IF numero MOD 3=0:co3=co3+1
6220 AT 11,30:PRINT co$
6230 AT 12,30:PRINT pr$
6240 AT 13,30:PRINT mt$
6250 END IF
6260 END DEFine
6270 :
6280 DEFine PROCedure impr_juga
6290 PRINT#6;t25$:OPEN#25;ser1
6300 IF pret=0:PRINT#25;ta$\tb$\' ';t32$:PRINT#25
6310 PRINT#25\t22$!njug!t23$!numero!co$!pr$!mt$
6320 PRINT#25
6330 PRINT#25\t30$\t31$
6340 FOR r=1 TO a
6350 PRINT#25,CHR$(27);"D";CHR$(1);
6360 PRINT#25,CHR$(13-LEN(men(r)));
6370 PRINT#25,CHR$(23-LEN(cnt(r)));
6380 PRINT#25,CHR$(33-LEN(gan(r)));
6390 PRINT#25,CHR$(0);
6400 PRINT#25,CHR$(9);aps$(r);
6410 PRINT#25,CHR$(9);men(r);'-';may(r);
6420 PRINT#25,CHR$(9);cnt(r);
6430 PRINT#25,CHR$(9);gan(r)
6440 END FOR r
6450 PRINT#25;CHR$(10)
6460 pret=1
6470 END DEFine
6480 :
6490 DEFine PROCedure impr_esta
6500 PRINT#6;t25$
6510 OPEN#25;ser1
6520 PRINT#25;ta$\tb$\FILL$(' ',10);t33$\\
6530 PRINT#25;'    ';'0-';nro(0)
6540 FOR ie=1 TO 36
6550 IF ie<=9:PRINT#25;'    ';ie;'-';nro(ie),
6560 IF ie>9: PRINT#25;'   ';ie;'-';nro(ie),
6570 IF ie MOD 9=0:PRINT#25
6580 END FOR ie
6590 PRINT#25
6600 PRINT#25;'   ';t59$,,fal,INT(fal*pct)!'%',,
6610 PRINT#25;'   ';t55$;'1',co1,INT(co1*pct)!'%'
6620 PRINT#25;'   ';t60$,,par,INT(par*pct)!'%',,
6630 PRINT#25;'   ';t55$;'2',co2,INT(co2*pct)!'%'
6640 PRINT#25;'   ';t61$,,roj,INT(roj*pct)!'%',,
6650 PRINT#25;'   ';t55$;'3',co3,INT(co3*pct)!'%'
6660 PRINT#25;'   ';t62$,,neg,INT(neg*pct)!'%',,
6670 PRINT#25;'   ';t57$;'1',do1,INT(do1*pct)!'%'
6680 PRINT#25;'   ';t63$,,imp,INT(imp*pct)!'%',,
6690 PRINT#25;'   ';t57$;'2',do2,INT(do2*pct)!'%'
6700 PRINT#25;'   ';t64$,,pas,INT(pas*pct)!'%',,
6710 PRINT#25;'   ';t57$;'3',do3,INT(do3*pct)!'%'
6720 PRINT#25:PRINT#6;t3$
6730 END DEFine
6740 :
6750 DEFine PROCedure cambiar (h)
6760 IF h=0:texto_3
6770 opc=CODE(INKEY$(-1))
6780 SELect ON opc
6790 =236-h:borrar_variables
6800 IF h=0:texto_4:sistema_Leigh
6810 IF h=4:texto_4:apuestas_libres
6820 =240:impr=1:PRINT#6;t20$:PAUSE 99
6830 PRINT#6;t3$:cambiar h
6840 =244:impr=0:PRINT#6;t24$:PAUSE 99
6850 PRINT#6;t3$:cambiar h
6860 =248:ett=1:estadisticas:cambiar h
6870 =REMAINDER :IF h=0:CLS#3
6880 END SELect
6890 END DEFine
6900 :
6910 DEFine PROCedure estadisticas
6920 CLS#4:fl=1:cl=2
6930 IF njug=0:PRINT#4;\\\\t34$:RETurn
6940 IF nro(0)>0:PRINT#4;t33$,'0-';nro(0);t33$
6950 IF nro(0)=0:PRINT#4;t33$,'0-';' ';t33$
6960 FOR cs=1 TO 36
6970 AT#4;fl,cl-LEN(çs):PRINT#4;cs;'-'
6980 IF nro(cs)>0
6990 AT#4;fl,cl+3-LEN(nro(cs)):PRINT#4;nro(cs)
7000 ELSE
7010 AT#4;fl,cl+2:PRINT#4;' '
7020 END IF
7030 fl=fl+1:IF fl MOD 7=0:fl=1:cl=cl+6
7040 END FOR cs
7050 neg=njug-roj-nro(0)
7060 imp=njug-par-nro(0)
7070 fal=njug-pas-nro(0)
7080 pct=100/njug
7090 PRINT#4;t35$
7100 PAUSE
7110 FOR li=1 TO 7:AT#4;li,0:PRINT#4;FILL$(' ',35)
7120 AT#4;1,0
7130 PRINT#4;' ';t59$,,fal,INT(fal*pct)
7140 PRINT#4;' ';t60$,,par,INT(par*pct)
7150 PRINT#4;' ';t61$,,roj,INT(roj*pct)
7160 PRINT#4;' ';t62$,,neg,INT(neg*pct)
7170 PRINT#4;' ';t63$,,imp,INT(imp*pct)
7180 PRINT#4;' ';t64$,,pas,INT(pas*pct)
7190 FOR lt=1 TO 6:AT#4;lt,29:PRINT#4;'%'
7200 PRINT#4;t35$
7210 PAUSE
7220 FOR li=1 TO 7:AT#4;li,0:PRINT#4;FILL$(' ',35)
7230 AT#4;1,0
7240 PRINT#4;' ';t57$;'1',do1,INT(do1*pct)
7250 PRINT#4;' ';t57$;'2',do2,INT(do2*pct)
7260 PRINT#4;' ';t57$;'3',do3,INT(do3*pct)
7270 PRINT#4;' ';t55$!'1',co1,INT(co1*pct)
7280 PRINT#4;' ';t55$!'2',co2,INT(co2*pct)
7290 PRINT#4;' ';t55$!'3',co3,INT(co3*pct)
7300 FOR lt=1 TO 6:AT#4;lt,29:PRINT#4;'%'
7310 PRINT#4;FILL$('+',35):PRINT#6;t21$
7320 REPeat seleccion
7330 slc=CODE(INKEY$(-1))
7340 SELect ON slc
7350 =240:PRINT#6;t20$:PAUSE 99:PRINT#6;t3$
7360 PAUSE:impr_esta:EXIT seleccion
7370 =244:PRINT#6;t24$:PAUSE 99
7380 PRINT#6;t3$:EXIT seleccion
7390 = REMAINDER
7400 END REPeat seleccion
7410 END DEFine
7420 :
7430 DEFine PROCedure texto_esp
7440 ta$= '      Q  L  -  R  U  L  E  T  A      '
7450 tb$= '         Por  Santiago Rubio         '
7460 t1$= '    H  A  G  A  N     J  U  E  G  O   '
7470 t2$= 'ESPERE 15 SEGUNDOS PARA CARGAR TABLAS'
7480 t3$= 'PRESIONAR CUALQUIER TECLA PARA SEGUIR'
7490 t4$= '       ELIJA LA OPCION DESEADA       '
7500 t5$= 'SITUE SU APUESTA CON CURSOR + <ENTER>'
7510 t6$= 'IMPORTE DE LA APUESTA = NUM DE FICHAS'
7520 t7$= 'LA APUESTA ESTA LIMITADA A '
7530 t8$= 'APUESTA INVALIDA.  QUEDAN '
7540 t9$= ' FICHAS'
7550 t10$='  ULTIMOS NUMEROS QUE HAN APARECIDO  '
7560 t11$=' INDIQUE CUANTO DINERO DESEA JUGARSE '
7570 t12$=' PARA GIRAR LA RULETA PRESIONE <ESP> '
7580 t13$=' LASTIMA.  HA PERDIDO TODO SU DINERO '
7590 t14$='  ENHORABUENA. HIZO SALTAR LA BANCA  '
7600 t15$='     DEBERIA INTENTARLO OTRA VEZ     '
7610 t16$='  NO LE QUEDA DINERO PARA CONTINUAR  '
7620 t17$='          N O   V A   M A S          '
7630 t18$='    CUANTOS FONDOS TIENE LA BANCA    '
7640 t19$='            SISTEMA LEIGH            '
7650 t20$='SITUE EL PAPEL Y CONECTE LA IMPRESORA'
7660 t21$='IMPRIMIR RESULTADOS: <F3> SI, <F4> NO'
7670 t22$=' Jugada:'
7680 t23$=' Numero:'
7690 t24$='         ANULADA IMPRESION           '
7700 t25$='       IMPRIMIENDO  RESULTADOS       '
7710 t26$='   FONDOS DEL JUGADOR:'
7720 t27$='    (Min:  $1000  Max:  $10000)     '
7730 t28$='   FONDOS DE LA BANCA:'
7740 t29$='    (Min: $10000  Max: $100000)     '
7750 t30$='  Suertes  Grupo Apuesta Ganancia   '
7760 t31$=' --------- ----- ------- --------- '
7770 t32$='  LISTADO RESUMEN DE LAS JUGADAS    '
7780 t33$='  Estadisticas'
7790 t34$=' No hay estadisticas disponibles.'
7800 t35$=' Mas informacion pagina siguiente..'
```

```
7810 t36$=' ------- Menu ------- '
7820 t37$=' <F1>-Apuestas        '
7830 t38$=' <F2>-Sistema Leigh   '
7840 t39$=' <F3>-Activa impresora'
7850 t40$=' <F4>-Anula impresora '
7860 t41$=' <F5>-Estadisticas    '
7870 t42$=' ---- Informacion --- '
7880 t43$=' Apostado.....:'
7890 t44$=' Recuperado...:'
7900 t45$=' Resto Jugador:'
7910 t46$=' Banca........:'
7920 t47$=' Jugadas......:'
7930 t50$='PLENO     ' :  t51$='CABALLO   '
7940 t52$='TRANSVSAL' :  t53$='CUADRO    '
7950 t54$='SEISENA   ' :  t55$='COLUMNA   '
7960 t56$='DOBLE COL' :  t57$='DOCENA    '
7970 t58$='DOBLE DOC'
7980 t59$='FALTA' : t60$='PAR   ' : t61$='ROJO '
7990 t62$='NEGRO' : t63$='IMPAR' : t64$='PASA '
8000 i0$='   Suertes   Numeros Limite Ganancia'
8010 i1$='  ---------  ------- ------ --------'
8020 i2$='  Pleno          1     20   35 veces'
8030 i3$='  Caballo        2     40   17    "  '
8040 i4$='  Transversal    3     60   11    "  '
8050 i5$='  Cuadro         4     80    8    "  '
8060 i6$='  Seisena        6    120    5    "  '
8070 i7$='  Columna        12   240    2    "  '
8080 i8$='  Docena         12   240    2    "  '
8090 i9$='  Ap.sencillas   18   360    1    "  '
8100 j0$='  Doble columna  24   480   1/2   "  '
8110 j1$='  Doble docena   24   480   1/2   "  '
8120 j2$='                                     '
8130 j3$='                 ----                '
8140 j4$='                                     '
8150 j5$='  Si sale el 0 se devuelve la mitad '
8160 j6$='    de lo apostado en las suertes    '
8170 j7$='              sencillas              '
8180 k0$='         Explicacion del juego       '
8190 k1$='-------------------------------------'
8200 k2$=' El jugador se enfrenta a una ruleta'
8210 k3$='americana, igual a las existentes en'
8220 k4$='un casino.   Las apuestas se colocan'
8230 k5$='moviendo las teclas de cursor y pre-'
8240 k6$='sionando <enter> una vez elegida  la'
8250 k7$='suerte que quiere apostar;  decida a'
8260 k8$='continuacion la cantidad y siga  las'
8270 k9$='indicaciones de la ventana inferior.'
8280 l0$=' El programa incluye un proceso para'
8290 l1$='apostar con el  sistema descrito por'
8300 l2$='"Norman Leigh"  en  su libro  "Trece'
8310 l3$='Contra La Banca"  y  que consiste en'
8320 l4$='aumentar  o  disminuir metodicamente'
8330 l5$='las apuestas que  6  jugadores hacen'
8340 l6$='a todas las suertes sencillas.      '
8350 l7$='                 ----                '
8360 END DEFine
8370 :
8380 DEFine PROCedure texto_ing
8390 ta$= '   Q  L  -  R  O  U  L  E  T  T  E   '
8400 tb$= '          By  Santiago  Rubio        '
8410 t1$= '       PUT  DOWN  YOUR  STAKES       '
8420 t2$= ' WAIT 15 SECONDS FOR  ARRAYS TO LOAD '
8430 t3$= '      PRESS ANY KEY TO CONTINUE      '
8440 t4$= '          MAKE YOUR CHOICES          '
8450 t5$= 'PLACE BETS USING  CURSOR PLUS <ENTER>'
8460 t6$= '     BETS AMOUNTS = $ 1 PER UNIT     '
8470 t7$= ' THE OPTION IS LIMITED TO '
8480 t8$= ' INVALID BET. IT REMAINS '
8490 t9$= ' DOLLARS'
8500 t10$=' THESE ARE THE LAST NUMBERS APPEARED '
8510 t11$=' SET THE AMOUNT YOU HAVE FOR BETTING '
8520 t12$='PRESS <ESP>  FOR THE ROULETTE TO SPIN'
8530 t13$='   SORRY!  YOU LOST ALL YOUR MONEY   '
8540 t14$=' CONGRATULATIONS! YOU BROKE THE BANK '
8550 t15$='        YOU  SHOULD  TRY  AGAIN      '
8560 t16$=' YOU HAVE  NOT ANY MONEY TO CONTINUE '
8570 t17$='       N O   M O R E   B E T S       '
8580 t18$=' SET THE AVAILABLE FUNDS OF THE BANK '
8590 t19$='              LEIGH PATTERN          '
8600 t20$=' SET THE PAPER AND SWITCH ON PRINTER '
8610 t21$='  PRINT RESULTS?: <F3> YES, <F4> NO  '
8620 t22$=' Throw:'
8630 t23$=' Number:'
8640 t24$='            PRINTER OFF             '
8650 t25$='          PRINTING  RESULTS         '
8660 t26$='    GAMBLER FUNDS:  '
8670 t27$='    (Min:  $1000  Max:  $10000)     '
8680 t28$='    BANK FUNDS:  '
8690 t29$='    (Min: $10000  Max: $100000)     '
8700 t30$='    Lot    Group  Stack    Return   '
8710 t31$=' --------- ----- ------- --------- '
8720 t32$='   PRINTED SUMMARY OF THE THROWS    '
8730 t33$='   Statistics'
8740 t34$=' Statistics are not yet available  '
8750 t35$='   More information next page...   '
8760 t36$=' ------- Menu ------- '
8770 t37$=' <F1>-Bets            '
8780 t38$=' <F2>-Leigh Pattern   '
8790 t39$=' <F3>-Printer on      '
8800 t40$=' <F4>-Printer off     '
8810 t41$=' <F5>-Statistics      '
8820 t42$=' ---- Information --- '
8830 t43$=' Staked.......:'
8840 t44$=' Returned.....:'
8850 t45$=' Gambler rest.:'
8860 t46$=' Bank funds...:'
8870 t47$=' Throws.......:'
8880 t50$='FULL      ' :  t51$='HORSE     '
8890 t52$='TRNSVERSE' :  t53$='SQUARE    '
8900 t54$='HLF DOZEN' :  t55$='COLUMN    '
8910 t56$='DBL COLUM' :  t57$='DOZEN     '
8920 t58$='DBL DOZEN'
8930 t59$='SHORT' : t60$='EVEN  ' : t61$='RED   '
8940 t62$='BLACK' : t63$='ODD   ' : t64$='PASS '
8950 i0$='    Lots      Numbers Limits  Return '
8960 i1$='  ---------  ------- ------ --------'
8970 i2$='  Full           1     20   35 times'
8980 i3$='  Horse          2     40   17    "  '
8990 i4$='  Transverse     3     60   11    "  '
9000 i5$='  Square         4     80    8    "  '
9010 i6$='  Half dozen     6    120    5    "  '
9020 i7$='  Column         12   240    2    "  '
9030 i8$='  Dozen          12   240    2    "  '
9040 i9$='  Single lots    18   360    1    "  '
9050 j0$='  Double column  24   480   1/2   "  '
9060 j1$='  Double dozen   24   480   1/2   "  '
9070 j2$='                                     '
9080 j3$='                 ----                '
9090 j4$='                                     '
9100 j5$=' If "0" appears the bank retuns half'
9110 j6$=' of stacks placed to  all the single'
9120 j7$='                lots                 '
9130 k0$='    Brief explanation of the game    '
9140 k1$='-------------------------------------'
9150 k2$='The gambler  faces up to an american'
9160 k3$=' roulette like  if he was in one of '
9170 k4$='those exciting casinos in Las Vegas.'
9180 k5$='   The bets are set using cursor    '
9190 k6$=' keys and pressing <enter> once the '
9200 k7$='choice has been made,  then put down'
9210 k8$='   the stack.  The process can be   '
9220 k9$=' repeated up to six times following '
9230 l0$='  the instructions in the windows.  '
9240 l1$=' The program  gives a tip including '
9250 l2$='      the pattern described by       '
9260 l3$='      Norman Leigh in his book       '
9270 l4$='     Thirteen Against The Bank,      '
9280 l5$='to increase or reduce the  bets that'
9290 l6$='six players place  simultaneuosly to'
9300 l7$='          all the single lots.       '
9310 END DEFine
9320 REMark ********   FIN DEL PROGRAMA   ********
10000 :
10010 REMark ************* BOOT PROG *************
10020 WINDOW 512,256,0,0
10030 MODE 4:CLS:CSIZE 3,1
10040 FOR a=200 TO  48 STEP -2
10050 CURSOR 96,a:PRINT '+------------------+'
10060 END FOR a
10070 FOR b=200 TO  68 STEP -2
10080 CURSOR 96,b:PRINT '|+----------------+|'
10090 END FOR b
10100 FOR c=200 TO  88 STEP -2
10110 CURSOR 96,c:PRINT '||      QL        ||'
10120 END FOR c
10130 FOR d=200 TO 108 STEP -2
10140 CURSOR 96,d:PRINT '||     RULETA     ||'
10150 END FOR d
10160 FOR e=200 TO 128 STEP -2
10170 CURSOR 96,e:PRINT '||    ROULETTE    ||'
10180 END FOR e
10190 FOR f=200 TO 148 STEP -2
10200 CURSOR 96,f:PRINT '|| SANTIAGO RUBIO ||'
10210 END FOR f
10220 FOR g=200 TO 168 STEP -2
10230 CURSOR 96,g:PRINT '|+----------------+|'
10240 END FOR g
10250 FOR h=200 TO 188 STEP -2
10260 CURSOR 96,h:PRINT '+------------------+'
10270 END FOR h
10280 CSIZE 0,0
10290 MERGE mdv1_ROULETTE
10300 RUN
10310 REMark ************************************
```

MICRODRIVE

THE PROGRAMS

Author	Language	Program Name	Price
1. Giles Todd	B	DIY Assembler	£5

Feaured in the March to June 1985 issues of *QL User*, this complete two-pass assembler will assemble all 68008 code and support the assembler directives DRG, END, EQU, DC and DS.

2. Richard Cross	A+O	Mini Monitor	£3

Using approximately 3K of RAM, this handy utility will multi-task on your QL, leaving plenty of room for other programs. Commands include dumping registers, memory — and ASCII — machine code trace, register store, memory move, memory store — byte, word and long — and jumps. Featured in *QL User*, October 1985.

3. A Didcock	B	Connect4	£1

A SuperBasic version of the classic four-in-a-row game where counters drop down slots in the vertical board. First printed in *QL User*, September 1985.

4. Shergold & Tose	B	Golf	£2

With up to 50 courses of varying difficulty, lakes, rivers, bunkers and trees, this is a fine golf simulation. You decide the power and direction of each stroke, striving for a birdie, eagle or even an albatross. Your scorecard may be saved. This program was printed in the May 1985 issue of *QL User*.

5. Williams & Holliday	A+O	Paladin	£5

Written completely in machine code, this excellent *Space Invaders* game was the basis of our games programming series, started in April 1985.

6. Richard Cross	M+B	Sprite Animation	£2

This contains two programs from the April 1985 issue. The first is a SuperBasic multi-coloured sprite designer. The second contains machine code routines to animate the sprites on the screen.

7. Steve Deary	B	Pacman	£1

A well-written maze game from the March 1985 issue. Almost 20 screens of increasing difficulty, including an invisible maze, make it a very versatile rendition of the arcade favourite.

8. Andy Carmichael	B	Family Tree	£3

Based on an article in the August 1985 issue, this is an Archive program and database for setting-up and displaying large family trees.

9. James Lucy	L	Composer	£3

Completed in *QL User*, October 1985 this QLiberated program will allow you to compose, play and amend your own melodies. The program will handle sharps, vary tempo, and even specify staccato and legato playing styles.

10. Matthew Capp	B	Miners	£2

This interesting simulation, printed in the August 1985 issue, puts you in the role of the NCB, buying and selling coal and mines, hiring and firing miners, and raising or decreasing wages to match economic forces. The object is to be profitable but inexperienced players will find it difficult even to remain solvent.

11. P J Smith	B	DIY Adventure	£1

From the February 1985 issue, this skeleton program requires you to slot in the details to create your own adventure programs.

12. R Green	B	Othello	£1

This classic board game, printed in *QL User*, August 1985, can be played by one or two players. The display uses a 3D representation of the board. Average response time by the computer opponent is about 15 seconds.

13. S J Ackers	S	Touch Type	£4

This program consists of a 13-lesson course for typing-in letters, words and phrases, a 700-word vocabulary, an interactive keyboard display and a fingering chart inas more than 30K of code. Scores are displayed based on the time and accuracy of typing. A reduced version of the program was printed in the August 1985 issue.

14. Rob Sherratt	A+O	Fcopy	£4

The first part of this program was printed in the March 1986 issue of *QL World*. The program is an ultra-fast, general-purpose file spooler.

15. Alan Prior	B	World Map	£2

From the March 1986 issue, this program will draw a full-screen, multi-coloured map of the world for geography buffs.

16. J M Dower	B	Mushyman	£2

Printed in the June and July 1986 issues, this provides speedy SuperBasic arcade action as you munch your way round the screen.

17. Tony Quinn	S	CAD QL	£4

CAD design programs are particularly suited to the QL. This version from the September 1986 issue includes features such as rubber-banding and a user-definable symbol library.

18. Stuart Campbell	M+B	Attack of the Things	£3

Typical science fiction horror arcade action as yet more nasties descend on harmless QL owners. Featured in the October 1986 issue of *QL World*.

19. Karl Jeffery	M+B	Starport 2001	£3

Fast machine code action in this November 1986 version of the *Galaxians* arcade game.

20. Marcus Jeffery	S	QL Go	£4

The oriental game of Go is so complex that even mainframe programs are easily beaten by novice players. To the best of our knowledge, this 15x15 version from the April and May 1986 issues is the only one available for the QL.

21. J P Hartley	B	Britain	£2

Another program for geography buffs from the November 1986 issue of *QL World*. This is a round-Britain geography quiz.

22. KBG Judson	B	Darts	£2

Program of the Month from December 1986. This popular pub pastime requires good hand and eye co-ordination to stop a moving cursor on the on-screen board.

EXCHANGE

KEY					
	B	= SuperBasic	A+B+O	= Assembler and Basic Loader and Object Code
	A+O	= Assembler and Object Code	S	= Supercharged
	M+B	= Machine Code and Basic Loader	L	= QLiberate

23. Neil Taylor S **Window Designer** £2

This useful routine from the February, 1987 issue allows you to design your SuperBasic windows using the cursor keys. It creates a one-line procedure which, when merged into your program, will re-create that window.

---

24. J F Tydeman S **Design 3D** £4

Published in the March and April 1987 issues, this program will allow you to produce 3D screen designs with the minimum of fuss and aggravation.

---

25. D Carmona B **Stellaris** £4

Program of the Month from June 1987. This is an extensive real-time space adventure game against the computer, including economic simulations, lunar landing and superb graphics.

---

26. Robert Noble A+B+O **Video Effects Box1** £3

These machine code SuperBasic extensions allow you to manipulate your screens, save and recall them from memory and clear them in interesting ways. Program of the Month for July 1987.

---

27. H R Pendry B **Pontoon** £3

A graphic version of the classic card game. You play against the computer. Features include changing banker on royal pontoons, accurate betting, five card tricks and so on. Printed in the July 1987 issue of *QL World*.

---

28. Kenneth Cameron B **Picture Puzzle** £2

This short but interesting program from the July 1987 issue sets up an 8 x 8 sliding block puzzle with on-screen graphics. You can select sliding numbers or load your own picture to solve.

---

29. Peter Etheridge B **Bridge** £4

An excellent version of this popular card game. Features include accurate computer bidding, automatic or manual play, replay hands, correct scoring, save and load positions and much more. Essential for card enthusiasts.

---

30. Charles Gerrard B **Psycho** £4

Based on an article in the July 1987 issue, this is an excellent version of the famous *Eliza* program. The cartridge contains a script design program, a pre-prepared script containing more than 50 keywords and an application program. Though written in SuperBasic, complex list processing makes this version extremely fast.

---

31. B Otridge **Crossword** £5

Sold originally as a commercial program, this is the perfect aid for crossword fanatics. The program provides access by word length to a dictionary of about 12,500 words, to help solve those elusive crossword clues. Note: This program requires two Microdrive cartridges.

---

32. Phillip Sproston B **Advent2** £4

SuperBasic arcade adventure with a humorous slant. A variety of rooms, robots and problems will keep you on your toes. Full instructions included.

---

33. Leslie Fahidy B **Clock** £3

This is a complete version of the clock program, described in the June and July 1987 issues of our QL Education series. An on-screen clock can be used to set or read the time.

---

34. E. Bamber **QL Conversion/ Calculator** £2

Comprising weights and measures units conventions and reverse Polish calculation, this excellent utility will convert almost anything to anything. Completely menu-driven, it is very easy to use.

---

35. John Wakefield B **Qwhist** £3

Our August 1987 Program of the Month. It is an excellent implementation of the classic card game, Whist. Designed for one player (south) who partners a computer hand (north) against the computerised east and west opponents.

---

36. Stanley Sykes B **Mail Merge** £1

This cartridge contains very handy utilities providing a mail merge and labeller for Quill files. The cartridge includes a simple demonstration.

---

37. P.G. Ives B **The Double** £4

A large strategy game in which you manage a football team through the four league divisions. The program features buying and selling, team line-up, morale, and so on, through the full league and F.A. Cup season. The cartridge includes full instructions Quill document showing how to play the game.

---

38. Leslie Fahidy B **Education** £2

As part of our series of educational programs, this is designed to help teach the solution of simple linear equations. It is aimed specifically at the 11-plus age range.

---

39. J F Tydeman S **Design 3D** £4

Featured in the March, 1987 issue, this extensive program includes a large suite of graphics and filing utilities for the production of 2D and 3D graphics. It is supplied complete with instructions in the form of a Quill document.

---

40. Santiago Rubio B **Roulette** £3

Our September, 1987 Program of the Month, this is an excellent Spanish/English version of the traditional gambling game. It also includes the Leigh Pattern, a system to break the bank.

---

41. Leslie Fahidy B **Money** £2

Continuing with our series of educational programs, this one sends you on a shopping expedition, calculating prices from shopping lists and trying to determine what coins you will receive as change.

---

42. Neil Davidson A+B+O **Life** £2

A machine code version of the classic simulation of a colony of living cells which survive, reproduce or die according to mathematical rules. Quill instructions included.

See over page for order form.

InStant

HARDWARE

ABC Electronic
010 49 521 8980881
Mouse/Software

Astracom
0792 473697
Modem

Care Electronics
0923 672102
Software and hardware specialist. Sandy official distributor. Q Jump official distributor. Wide range of leads available.

Chromagraphica
Colour monitor, high resolution, low cost. Full details from Chromographica, 135 Cliff Road, Hornsea, North Humberside HU18 1JB or telephone 04012 4699.

Computer Supplies
146 Church Road, Boston, Lincs.
Joysticks

Compware
0270 582301
Disc Drives, Memory, Monitors/Software

CST
24 Green Street, Stevenage, Herts SG1 3DS. Tel: (0438) 352150
Supplies of QL storage devices including QDisc. Interface and Ram drives 3.5in./5.25in., Dual 720K 3.5in. floppy disc drives. Ram plus 512K expansion QDisk and dual floppy, 20 MB Winchester, Q488 IEEE GPIB interface, QEP-111 Eprom programmer. Also supplies of Thor 1F, 2F, WF. Telephone for details.

Cumana Ltd
0483 503121
Disc Drives/Interface

Data Distributors Ltd
0990 28921
Kaga, Sinclair Vision

Datalink
031 554 6040
Printer interfaces

D. J. W. Software
11 Pound Close, Bramley, Hampshire, RG26 5BL

Eidersoft
0708 851099
QL Mouse, Disc Drives/Software

Eprom Services
0532 667183

Tony Firshman Services
01-724 9053
Mains spike filter

4 Systems
01-844 1399
Hardware and software

Kempston Ltd
0234 856633
Disc/Centronics interface

Management Science Ltd
17 West Hill, London, SW18
QL Case/Software

Micro Peripherals Ltd
0256 473232
Disc Drives/interfaces

Microworld
0293 545630/0273 6711863
Printers and monitors

Mirace Systems
0272 603871
Expanderam 512K/Software

Opus Supplies Ltd
Redhill 65080

PCML Ltd
0372 67282/68631
QL & RAM cards/Software

Penman Products Ltd
0903 209081
Penman plotters

Sandy (U.K.) P.C.P.
0234 219814
QLT, Disc systems, SuperQboard, RAM boards

Silicon Express
0533 374917
Disc interface/drives, Centronics interface, RAM boards, Monitors, Printers

Slave Software
050 846 8866
Q Disc/Software

SMC Supplies
01-441 1282
Joystick adaptor, Centronics and Epson Serial

Strong Computer Systems
0267 231246
Printers and Monitors/Software

T.K. Computerware
0303 812052
Stockist

Tandata
06845 68421
Q-Connect, Q-Mod, Q-Call

Technology Research Ltd
0784 63547
Disc interface with RAM and Centronics port

Transform Ltd
089 283 4783
QL dust cover, Microdrive storage box, RS232 lead, disc drives, monitors and printers

Viglen Computer Supplies
01-8112 9902

SOFTWARE

Adder Publishing Ltd
0223 277050
Utilities

Ark Distribution
Corve Farm House, Chale Green, Ventnor PO38 2LA. Tel: 098-379 496
Specialists in Archive systems for QL/PC/OPD computers and File Conversion/Recovery/Transfer Services. Our systems include Archivist (£29.95) a powerful all purpose text manager, Card File (£14.95/£19.95). A specialist addressing system, Mail File (£14.95/£19.95). A general addressing system with notes, Librarian (£19.95/£24.95), for cataloguing items like books, articles, records and videos, and Mail Merge (multi tasking) for producing quantities of formated reports. Using an archive export file and a quil Lis file.

Breakthrough Software
0763 45482
Utilities

Bridgebrook Intek
45 Burleigh Avenue, Wallington, Surrey
Utilities

Business Software
01-863 1861
Utilities

Collins Soft
Freepost, 30, London W1E 7JZ
Utilities

Consumers Association
Subscription Dept, PO Box 44, Hertford, SG14 1SH
Utilities

C.P. Software
10 Alexandra Road, Harrogate
Games

D. A. Bandoo
81 Mount Pleasant, Wembley
Utilities

D. J. Walker
22 Kimptons Mead, Potters Bar, Herts EN6
Utilities

D. S. Enterprises
01-671 0209
For the best deal in town on drives interfaces extra memory, software, discs, books and while stocks last used-once Microdrive cartridges. Our own software includes Ram Disk (world's fastest), sign designer and multi Task.

Data Management
0904 760351
Utilities

Datalink Systems
097081 360
Utilities & Games

Digital Precision
01-527 5493
Utilities & Games

D.R.K. Products Ltd
Bar 1, Pipers Lane, Markyate, St Albans, Herts AL3 8QF
Micro Cartridge Rack

D. S. Enterprises
01-671 0209
Utilities

Eigen Software
45 Bancroft Road, Widnes, Cheshire WA8 0LR
Games

Equate
2 Ffordd Derwyn, Penyffordd, Chester CH4 0JT
Utilities and games

Gemini Marketing
0395 265165
Utilites and games

G. S. T. Computer Systems
0954 81991
Utilities

Hisoft
0582 696421
Utilities

Intek Software
01-500 8534
Utilities

J. & D. Software
3 Alfred Road, Lowtown, Warrington

Key Software
0703 695182
Games and utilities

McGraw-Hill
0628 23431
Utilities

MetaComCo
0272 428781
Utilities

Microprocessor Engineering Ltd
Hanley Road, Southwater, Horsham, Sussex
Games and utilities

MicroApl
01-622 0395
Games and utilities

Microdeal
QL Software available from Microdeal PO Box 68, St Austell, Cornwall PL25 4YB. *The King £7.50 multi-screen Kong arcade game. Aquanaut 471 £14 graphic adventure/over 100 screens. Lands of Havoc £10 2000 screen graphic adventure. Cuthbert in Space £7.50 arcade zap em up. Crazy Painter £5 zany paint the floor game. Hopper £5 original Frogger-style arcade game*

N. Alexander
53 Allen Road, Wolverhampton, West Midlands
Utilities

New Horizons Software
Fourwinds, 30 Cwm Lane, Rogerstone, Gwent
Games

Owl Computer Training
044 282 7302
Utilities

Paddy Software
0726 832718
Utilities

Palantir Products
60 St Lukes Road, Bedminster, Bristol
Utilities

PCS
8 Oakgrove Way, Bridgewater, Somerset
PCS Utilities

Peakcrown Ltd
4 Beeby Road, Scraptoft, Leicestershire LE7 9SG
Games

Peak Electronics
32 Clifton Avenue, Hartlepool, Cleveland
Utilities

Plantir Products
60 St Luke St, Bedminster, Bristol
Utilities

Portfolio Software
PO Box 15, London SW11

Position Computing
0554 759624
Utilities

Prentice Hall
0442 58531
Utilities

Prospero Software
Pro Fortran-77 is a full Ansi Specification Fortran 77 compiler for the QL. Pro Pascal is a full ISO specification Pascal compiler for the QL. Both consist of a 200-page manual, ROM cartridge and three Microdrives, and run on unexpanded 128K QL. Price £99.95 including VAT from Prospero Software Ltd, 190 Castelnau, London SW13 9DH. *Enquiries and credit card orders to 01-741 8531.*

Psientific Software
0482 649187
Utilities

Psion
01-723 9408/0553
Games and utilities

Pyramide
01-349 2764
Games and utilities

QCode
0235 28359
Games and utilities

Qflash
Hamburg 6512742
Utilities

Qjump
0954 50800
Utilites

QSoft
01-559 7417
Utilities

Quest
04215 66488
Utilities

Realtime Software
0532 458948
Games

Rubicon Micros
0742 583665
Games and utilities

S. & B. Software
20 St Nicholas Street, Diss, Norfolk
Games

Sector Software
45 Cheetham Meadow, Moss Side, Leyland, Lancs PR5 3UB
Games and utilities

Shadow Games
1/2 The Cottages, Maidenhatch, Tidmarsh, Berks RG8 8HP
Games and utilities

Smiling Software
26 Dale Road, Marple, Stockport SK6 6HA
Games and utilities

Solent Systems
0983 865694
Utilities

Stevenson & Partners
0428 51500
Utilities

Super Plant Software
097 423 223
Utilities

Swansoft
164 Vicarage Road, Morriston, Swansea
Games

Talent Computer Systems
041 552 2128
Games and utilities

TDI Software Ltd
0272 742 796
Utilities

I wish to update my Instant Access entry.
Please contact me.

Name_____

Company _____

Telephone _____

Please return to Jane Brooks Wadham,
SQLW, Greencoat House, Francis Street, London SW1P 1DG.
Or telephone on 01-834 1717 ext. 260.

INSTANT ACCESS

TR Computer Systems
093 924 621
Utilities

WD Software
0534 81392
Utilities

ZitaSoft
Steve Jones, 93 Foxbourne Road, London SW17 8EN
Utilities

SERVICES

Computer Repair Centre
0379 52588

Data Printing Services
25 Seymour Road, Bath, BA1 6DY

Owl Computer Training
044 282 7302

Quanta
0908 564271

Rainbow Repairs
0904 768853

Zedem Computers Ltd
0234 213032

QL REPAIRS

ANY FAULT **£27.95** INCLUSIVE OF P&P, VAT INSURANCE AND 3 MONTHS GUARANTEE. FOR REPAIRS ON OTHER MICROS PHONE FOR DETAILS.

SUREDATA

Telephone: 01-902 5218

UNIT 6, STANLEY HOUSE
STANLEY AVENUE, WEMBLEY
MIDDX. HA0 4JBQ

DUMP TROUBLE?

TRY HYPERDUMP

Transfer selected screen areas
Construct posters up to 240 linear times
Multitasking with variable priority
Accurate screen proprtions maintained
Epson compatible printers

Further details from: HARVEY SOFTWARE, 30 EDIES LANE, LEAVENHEATH, COLCHESTER, ESSEX CO6 4PA. TEL: 0206 262912

OR send £9.95 specifying cartridge or 3.5" DSDD disk.

ERROR TRAPPING

At last, it's now possible to have real error Trapping in your SuperBASIC program.

Traps *all* errors from SuperBASIC extensions.

Amazing — it can even Trap errors from all TOOLKITS currently available or extensions you care to add to SuperBASIC!!!

Works with *ALL* versions of the QL — £6.

Utilising Software 75 Oakhanger Drive Lawrence Weston Bristol BS11 0RJ

*** SOCCER CHIEF + ANDROMEDA ***

Buy Soccer Chief and get Andromeda FREE

SOCCER CHIEF — a comprehensive football management game. League + FA Cup, player/team ratings etc

ANDROMEDIA — graphic space adventure where you must battle and trade throughout the galaxy in a quest for the Golden Orb.

On microdrive for only £12 including p&p — overseas add £1

M McQuiod, PO Box 189, COVENTRY, CV6 3HL

FOR A PERMANENT SOLUTION TO POWER CUTS AND MAINS INTERFERENCE FIT AN UNINTERRUPTABLE BATTERY BACKUP TO THE QL

BACKUP INCORPORATES FLOAT CHARGED NICAD BATTERIES PERMANENTLY CONNECTED TO PRESERVE DATA IN THE QL PROCESSOR, MICRODRIVES AND RAM DURING MAINS INTERFERENCE. SUPPLIES THE QL DURING POWER CUTS FOR UP TO AN HOUR. NO SOLDERING OR OPENING UP NECESSARY AND REGULATOR REDUCES QL OVERHEATING

£54 inc.

to FREQUENCY PRECISION LTD Hillsview, Aller Rd, Dolton, Devon EX19 8QP. Tel. 0805 4381

NEW SOFTWARE FOR QL, SANDY, CTS THOR

AUTOSPELL — Spell corrector (not just a checker) — multilingual 29£

FILE MANAGER — Sophisticated file management for copies, deletions, comparisons, display, "Wild card" selection — Handles all types of files. Multilingual 22£

Distributors Desired

SOFTSPEED sprl, 90 rue St Lambert, 1200 Brussels, BELGIUM tel: (322) 762.69.46)

curry computer

Your Complete Sinclair Stockist

Hardware, Software, Peripherals,
Printers, Books, Magazines, Accessories

P.O. Box 5607 • Glendale, AZ 85312-5607 U.S.A.
1-602-978-2902 • Telex (via WUI): 6501267701

DEALER INQUIRIES WELCOME

PROBLEMS WITH YOUR QL!

FOR FAST, RELIABLE AND PROFESSIONAL REPAIRS AT COMPETITIVE PRICES

CONTACT:-

PM ENGINEERING

UNIT 8, NEW ROAD, ST IVES, CAMBRIDGESHIRE. PE17-4BG

ST IVES (0480)61394

MICRO ADS

"NEW RIBBONS FOR OLD" from ALADDINK

ALADDINK WILL RE-INK YOUR OLD FABRIC RIBBONS AT GREAT SAVINGS AGAINST NEW PRICES!!

INTRODUCTORY OFFER

Send a dry fabric ribbon cassette (stating printer make and model) with £2 and we'll re-ink it and quote you for future re-inkings.

ALADDINK (Dept QL), 4 Hurker Crescent, Eyemouth, Berwickshire TD14 5AP. Or 'phone for a quotation: **08907 50965**

Q.L. Educational

SPELL

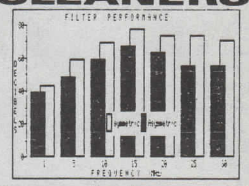
A suite of first programs for 2 to 7 year olds. Teaches use of keyboard, letter recognition and uses pictures to teach spelling.

EXTRA FREE "DRAW" A Graphics program for use by very young children.

Send a blank Microdrive and a cheque for £3— To: Mr A.C. Clarke—Williams, 3, Hammond Road, Basingstoke, Hants RG21 3NH.

Tony Firshman Services

COMPUTER CLEANERS



Many computer systems, disc drives, printers, modems, telex machines, Hi-Fi and CD systems are sensitive to 'dirty' power supplies. Voltage spikes or electromagnetic (RF) interference sent down power supplies when neighbouring equipment is switched or lightning strikes can crash or damage systems and corrupt data. We manufacture three products designed to overcome this problem — two adaptors (5a fuse) with one and three 3-pin sockets respectively and a 4-way socket (13a) with trailing lead (similar filtering circuitry). They can also reduce mains borne noise between computers and cordless phones/FM intercoms.

SPECIFICATION

1. *Capacitive* Delta filter to BS 2135. L/N capacity boosted to over 430n with a discharge resitor to prevent shocks.
2. *Inductive* EMI RF chokes rated to 130MHz used in series. Clipped voltage spikes smoothed.
3. *Spike suppressors* (VDR) Three fitted (L/N, L/E, N/E) to short voltages above 630v. (peak current .1200a. 130 joules dissipation. Response time < 10 Ns) (only 1 in 1-way).

All circuits are in standard square pin devices and are plug in — NO WIRING REQUIRED.

1-way adaptor £14
3-way adaptor £18
4-way trailing socket £24
PRICES ARE FULLY INCLUSIVE

VISA

"The cleaner has been in operation for a month with my QL and I am glad to report that locking up problems appear to be a thing of the past. Many thanks for your help, efficiency and, above all, for producing such an effective device" — N Daglish, New Zealand.

Supplier to the DHSS, MOD, Universities, Schools, Hospitals and Police Authorities

12 Bouverie Place, London W2 1RB. Tel: 01-724 9053
Telex: 265451 MONREF G (quoting BT Gold 72: MAG90645)

TAXAN/KAGA AND CANON PRINTER OWNERS

(TAXAN KP810, KP910 and CANON PW1080A, PW1156A)

IF YOU WOULD LIKE THE HIGHEST QUALITY PRINTER OUTPUT, THEN WRITE OR PHONE FOR A (FREE) 7 PAGE DEMONSTRATION OF **"CuePrint"** with **"Printer Toolkit"**

MUST BE SEEN TO BE BELIEVED!

D. A. BURRIDGE, 23 CROMWELL AVENUE, HAMMERSMITH, LONDON W6 9LA. TEL: 01-748 3437

COPE ANNOUNCES THE END OF THE ICE AGE!

QATS Version 3 is the front-end system with more facilities and more scope for expansion. Powerful enough to control the largest QL's but compact enough to operate in an unexpanded QL. (QATS + 16K EPROM leaves 40K for the user.) QATS grows with you as your QL and program library grow, and when you move up to a larger machine you wont have to change your operating methods. QATS runs all EXEC-able and SUPERBASIC jobs.

QATS is menu-driven for speed and simplicity. Every menu is its own help screen and can be tailored by the user. See our reviews in SQLW Nov and Dec 1986. And August 1987.

QATS Version 3: £25.00; Output Control: £20.00; 16K EPROM: £15.00; 32K EPROM: £20.00. These prices include postage and media — MDV, 3.5 or 5.25 (state which).

Send for details. Available on 21 day approval. £5 off (QATS + Compware's SWOPPER).

COPE, 3 Langham Mansions, Earls Court Square, London, SW5 9UH

QL SMALL TRADERS PACK

New, enhanced QUICK-LOADING version 2! Accounts, Stock Control, Mailing List, Label Printer, Phone File, mini-calc plus utilities. "Simple to use, a good deal for the money." (QL WORLD) *Fully menu-driven & Epson-compatible* Supplied on two mdvs for only £24.95! (£10 update) QL INVOICER — prints smart invoices & statements on plain paper. Only £14.95 (£9.95 with Pack) QL BANK ACCOUNT £9.95 QL FILOFAX (4 progs) £9.95 Send SAE for full details:

SD MICROSYSTEMS, (DEPT QL) P.O. BOX 24 HITCHIN HERTS UK.

Format conversion between QL Microdrive, **PC/MS-DOS** and **CPM** disc formats

Conversion price £5·00 for 1, reducing depending upon number.

Tel for details:
PETER F. MANSELL LTD.
09946-370

You can advertise your own business in QL World for as little as £15 inc VAT for a minimum of 30 words. Send details along with your name, address and telephone number to Classified, Sinclair QL World, Greencoat House, Francis Street, London SW1P 1DG or ring Jane Brookes Wadham 01-834 1717.

MPC SOFTWARE

SOFTWARE

Leisure Software

Bridge Player	£12.00
Chess	£17.00
Cuthbert in Space	£10.00
Deathstrike	£13.00
Eidersoft 3 Pack	£13.00
Farmer	£13.00
Fictionary	£10.00
Flight	£17.00
Hopper	£9.00
Hoverzone	£13.00
Jabber	£9.00
Jungle Eddie	£13.50
Karate	£12.00
King	£10.00
Lands of Havoc	£10.00
Lost Pharoah	£13.00
Matchpoint	£13.00
Mr. Smith	£10.00
Othello 3D	£12.00
Oxford Trivia	£16.00
Quboids	£9.00
Scrabble	£13.50
Squadrons	£14.50
Startrek	£4.00
Steve Davies Snooker	£13.00
Stone Raider II	£10.00
Strip Poker	£13.50
Super Croupier	£13.50
Tank Busters	£15.00
Type 22 (Submarine Games)	£15.00
Vroom	£13.50
Wanderer	£17.00
War in The East	£PHONE

ADVENTURES

Aquanaut 471	£18.00
Dark Side of the Moon	£16.00
Horrorday	£13.00
Mortville Manor	£17.00
Nemesis	£13.50
Nemesis + Music	£17.00
West	£13.50
ZKul	£13.50

Business Software

Cash Trader	£55.00
Concept 3D (demo £2)	£25.00
Concept 3D + Viewpoint	£40.00
Decision Maker	£32.00
Entrepreneur	£34.00
Front Page	£21.00
Front Page Extra	£37.00
PCB Designer 1	£90.00
PCB Designer 2	£45.00
Project Planner	£32.00
Pro CAD 3D	£34.00
QRAM (Mdv - Disc)	£29.00
Sideways	£18.00
Spellbound (Mdv - Disc)	£29.00
Task Master (Mdv - Disc)	£23.00
Technikit	£24.00
Techniqf	£40.00
Transact (Book Keeping)	£20.00
Viewpoint	£17.00
Viewpoint disc	£22.00

Languages

Assembler (Metacomco)	£35.00
GST Assembler	£22.00
Assembler Workbench	£21.00
C	£80.00
Forth (computer one)	£30.00
Fortran 77 (Prospero)	£90
Macro Assembler (GST)	£37
Mon QL	£20.00
Pascal (Metacomco)	£74.00
Pascal (Prospero)	£90.00

Utilities

ArtICE	£12.00
BASIC - ally	£26.00
Bag of Tricks	£11.00
Cartridge Doctor	£13.00
Complete Toolkit	£12.00
Copier & Spooler	£7.00
Cryptor	£13.00
Font Designer	£7.00
Gardener	£12.00
Graphics Toolkit	£13.00
ICE ROM (New Version)	£27.00
Memory Shrink	£5.00
Nucleon	£17.00
QFlash ROM	£23.00
QFlash ROM (Cumana Disc Rom Rep.)	£23.00
QFlash RAMDisc & T oolkit	£18.00
QIMP	£24.00
RAM Disc	£14.00
Spell Bound (Mdv - Disc)	£29.00
Sprite Designer	£13.00
Task Master	£24.00
Touch Typist	£11.00

BOOKS

QL Book of Games	£4.00
Sinclair QL Adventures	£2.50
Artificial Intelligence on the QL	£3.00
Maths on the QL	£3.50
Developing Applications on the Sinclair QL	£3.00
Simulation Techniques on the Sinclair QL	PHONE
Quill, Easel, Archive & Abacus on the QL	PHONE
Inside the Sinclair QL (Hardware)	£6.00
Desk-top Computing with the Sinclair QL	£3.00
Word Processing with the Sinclair QL	£3.00
Database Management on the Sinclair QL	£4.00
Advanced Programming with the Sinclair QL	£4.00
Making the Most of the Sinclair QL	£3.00
Using Graphics on the Sinclair QL	£7.95
Introduction to Superbasic on the Sinclair QL	£3.50
Introducing the Sinclair QL	£3.50
Profiting from the QL (Abacus)	£4.00
QL Computing	PHONE
QL Advanced User Guide (QDOS)	£15.00
Quantum Theory	£4.00
QL Quill	£4.50
QL Easel	£4.50
Sinclair QDOS Companion	£6.95
Assembly Language Programming on the QL	PHONE
QL Service Manual (circuit diagram etc.)	£19.95
Hackers Handbook	£4.00
Creating Adventure Programmes on your Computer	£2.50
How to Computerise Your Business Sucessfully	£2.50
QL Handbook	£5.95
QL Compendium	£5.95
Exploring Artificial Intelligence on the QL	£3.50
Abacus Users Reference Manual	£5.00
Quill Users Reference Manual	£4.00
Computer Art & Graphics	£4.00
IBM PC Users Guide	£4.50
Xenix Users Handbook	£5.00
MS-DOS, an Introduction	£5.00

Sinclair QDOS Companion

The QDOS Companion, by Andrew Pennell, one of the most prolific authors on the QL, is essential reading if you are interested in programming the QL in Machine Code. It describes all the functions of QDOS, and gives details of all the TRAPs, and VECTOR calls. It includes details on how to access the keyboard, screen, serial ports etc. from machine code, how to communicate with the 8049 co-processor, and how to write multi-tasking programmes and lots more. £6.95

QL Service Manual

The QL Service Manual is the book issued by Sinclair Research to QL repair companies, and contains all you need to know about the QL Hardware. It has full circuit diagrams including the microdrives, details on various chips and their functions, block diagrams, fault finding and lots more £19.95

7 Book Pack

The following seven books from the Hutchinson QL range at less than a third of their original price (over £50).

Word Processing on the QL
Profiting from the QL
Introduction to Superbasic on the QL
Desk top computing on the QL
Database Management on the QL
Making the Most of the QL
Advanced Programming on the QL

All 7 now only £16

Using Graphics on the QL

This book has been written with the intention of introducing both beginners and experienced programmers to ways of creating graphics on the QL. Topics covered include:

Using graphics with Easel
Graphics from SuperBASIC
Windows
Turtle graphics
3D Graphics, and lots more . . . .

Only £7.95

HARDWARE

Disc Products

Cumans disc interface	£80.00
Interface & single drive	£200.00
Interface & Dual drive	££295.00
10x3.5" DSDD discette (Branded)	£25.00
40x3.5" Discette box	£3.00
90x3.5" lockable storage box	£12.00
3.5" discette cleaner	£8.00

Prices quoted for 3.5" systems.

Memory Products

Miracle	Expanderam95.00
Internal 512k upgrade	£100.00
Internal 512k upgrade fitted	£135.00
Q-Flash RAM disc	£18.00
Q-Flash RAM disc on rom	£25.00
Miracle Trump card	£195.00
Memory 'shrink' program	£5.00

Peripherals

Centronics printer interface	£20.00
Serial Cable	£7.00
QL Computer & Software	PHONE
Video Digitizer	PHONE
Keyboard Membrane	£8.00
Eidersoft ICE & Mouse	£65.00
Microdrive Cartridges	£1.80
4 Microdrive cartridges in wallet	£7.50
Joystick Adaptor	£4.50
Microvitec 1541DQ 3 colour monitor	£270.00
Miracle Modem	£49.00
Monitor Leads	PHONE
Schon Keyboard	£54.00
NEW SCHON keyboard	£95.00

Postage:

UK: FREE on orders over £10.
£1 on orders under £10.
Europe: £1 plus £1 per book.
Outside Europe: Write.

Printers

Citizen 120D
120 CPS draft, 25 CPS NLQ £210.00
Citizen MSP20
200 CPS draft, 50 CPS NLQ £450.00
Colour printers
& Laser Printers ..... PHONE

ribbons £5 each

Printers are supplied complete with cable etc.

Joysticks

Joystick Adaptor	£4.50
Elite	£13.00
Micro Ace	£15.00
Cheetah 125	£9.00
Pro Ace	£9.00
Pro 9000	£18.00
Professional standard	£15.00
Arcade	£19.00

All prices include postage & VAT

Also available a range of software & hardware for other 68000 machines (Amiga & ST).

Mail Order Address:

MPC Software
72 Julian Road,
West Bridgeford,
Nottingham NG2 5AN
Tel: 0602 820106

Personal callers come to:

The Nottingham 68000 Centre
40 Bridgeford Road,
West Bridgeford,
Nottingham NG2 6AP
Tel: 0602 455114

All prices correct at the time of going to press. E&OE.

STRONG COMPUTER SYSTEMS

BRYN COTTAGE, PENIEL, CARMARTHEN, DYFED SA32 7DJ. TEL: 0267-231246

THE ASTRACOM 1000 MULTISTANDARD INTELLIGENT MODEM £198.00
Complete with Communications Software * * NOW WITH BT APPROVAL * *
128K QL COMPUTER ..... £149.00 640 QL COMPUTER £239.00
SILVER REED PLOTTER WITH TALENT TECHNIKIT ONLY £199.00
QUALITY PRINTER STAND - Suitable for any 80 Col Printer ONLY..... £59.00
Small Footprint & Dial Control For Smooth Paper Feed
200 WATT UNINTERRUPTABLE POWER SUPPLY ONLY £229.00

* * FUTURA PLEASE CALL * *

NEW!

* * TONY TEBBY'S NEW PRODUCTS * *
QRAM: MULTITASKING, WINDOW CONTROL UTILITIES ETC. £29.00
QRAM + QUALITY MOUSE + INTERNAL MOUSE INTERFACE..... £89.00
QUALITY MOUSE + INTERNAL MOUSE INTERFACE £69.00

* * NEW FROM MIRACLE SYSTEMS * *
TRUMPCARD: THE NEW DISK INTERFACE + RAM UPGRADE FROM MIRACLE SYSTEMS INCLUDES 768K OF RAM UPGRADING YOUR QL TO A TOTAL OF 896K RAM + TOOLKIT II + MULTI MODE SCREEN DUMPS + RAM DISK & PRINT SPOOLER
TRUMPCARD WITH FREE QRAM ONLY..... £199.00

SPELL BOUND: SPELLING CHECKERS WITH 30,000 WORD DICTIONARY
£29.00 £29.00

NEW!

SUPER Q BOARD INCLUDES 512K + PARA PORT + TK2 WITH FREE QRAM £199.00
ABOVE WITH NEC DUAL DRIVES £399.00
NEC DUAL 3.5" DISK DRIVES £209.00
NEC DUAL DRIVES + SUPERDISK INTERFACE WITH PARA PORT £269.00
SUPERDISK INTERFACE INCLUDES PARA PORT ..... £ 79.00
SANDY IBM KEYBOARD UPGRADE..£79.00 SCHON KEYBOARD £57.00
10 DS/DD 3.5" DISK IN CASE.....£26.00 LOCKUP BOX FOR 40 3.5" DISKS £14.00
SPECIAL OFFER: 20 3.5"DISKS + FREE STORAGE FOR 40 DISKS ..... £ 52.00

SPECIAL OFFER: 6 BOOKS ONLY £19.00
THE QDOS COMPANION
ASSEMBLY LANGUAGE ON THE SINCLAIR QL
QL ADVENTURES
MATHS ON THE QL
ARTIFICIAL INTELLIGENCE
DEVELOPING APPLICATIONS
* * FREE WITH EACH QL COMPUTER * *

PRINTER PRICES INCLUDE MIRACLE SYSTEMS INTERFACE

SEIKOSHA 1300AI 300 CPS (colour option £100 extra) £399.00
EX DEMO PRINTERS AVAILABLE PHONE FORE LATEST PRICES
SEIKOSHA SP180 ..... £189.00
CANON PW1080 ..... £289.00
KAGA TAXAN KP910 WIDE CARRIAGE EX DEMO £350.00
DAISEYWHEEL MP26 (26 CPS) QUME COMPATIBLE..... £299.00
2000 Sheets Fanfold Paper Micro Perf ..... £ 19.00
2000 Sheets Fanfold Paper (supplied with printer) ..... £ 16.00
MIRACLE SYSTEMS PARALELL INTERFACE..... £ 20.00

ACCESSORIES

CLEANING KIT IN CARRYING CASE..... £ 27.00
3250 WATT MAINS FILTER ..... £ 36.00
MIRACLE SYSTEMS MODAPTOR ..... £ 39.00
MICROSTICK JOYSTICK WITH ADAPTOR ..... £ 16.00
20 CARTRIDGES WITH TRANSFORM BOX ..... £ 40.00
10 CARTRIDGES WITH TRANSFORM BOX ..... £ 23.00
4 Mircrodrive Cartridges £8.00
Joystick Adaptor £5.00
Serial Printer Cable..... £8.00
Transform Box..... £ 5.00
Para Cable for Memodisk £10.00
10 DS/DD 5.25" Disks... £19.00

MONITORS (Cables Included)

Microvitic Cub 1451/DQ Colour £249.00
Microvitic Cub 1451/DQT3 With Swivel Stand £269.00
Swivel Stand For Microvitic ..... £ 25.00
Phillips BM7502 20 MHZGreen (RECOMMENDED FOR THE QL)..... £ 99.00

RIBBONS

BROTHER M1009 .....£5.00
QUEN DATA £6.00
TAXAN/CANON £8.00
EPSON LX80 ..... £5.00
MT80 or Shinwa ..... £ 6.00
SP1000A (Sinclair) ..... £ 7.50
EPSON RX/MX/FX ..... £ 6.50
CITIZEN 120D ..... £ 6.50

Software

BUSINESS

INVESTMENT MONITOR ..... £ 19.00
PBC DESIGNER 1 ..... £ 89.00
PCB DESIGNER 2 ..... £ 89.00
TALENT TECHNIQL ..... £ 44.00
CONCEPT 3D CAD ..... £ 34.00
RUBICON VIEWPOINT ..... £ 24.00
DIGITAL EYE Q ..... £ 24.00
QDRAW ..... £ 14.00
SIGN DESIGNER ..... £ 19.00
FRONT PAGE ..... £ 24.00
PRO ASTROLOGER CART/DISK ..... £ 49.00
PRO ASTROLOGER + ASTRONOMER ..... £ 59.00
QL SERVICE MANUAL ..... £ 19.00
PROCAD 3 ..... £ 39.00

LANGUAGES

METACOMCO ASSEMBLER ..... £ 29.00
METACOMCO BCPL ..... £ 49.00
METACOMCO LISP..... £ 49.00
METACOMCO PASCAL ..... £ 69.00
METACOMCO 'C' ..... £ 79.00
PRO FORTRAN ..... £ 89.00
PRO PASCAL ..... £ 89.00
DIGITAL TURBO ..... £ 89.00
TALENT WORKBENCH ..... £ 19.00
QL LIBERATOR ..... £ 59.00
FORTH + REVERSI ..... £ 35.00

UTILITIES

DIGITAL MEDIA MANAGER ..... £ 35.00
THE EDITOR ON CART (128K VER.) .... £ 24.00
THE EDITOR ON DISK (640K VER.) ..... £ 39.00
TALENT TECHNIKIT ..... £ 24.00
TALENT QIMP ..... £ 29.00
TALENT BASIC ALLY ..... £ 29.00
DISK & PRINT SPOOLER ..... £ 9.00
TALENT SIDEWAYS..... £ 19.00
DIGITAL SPRITE GENERATOR ..... £ 29.00
LOCKSMITH ..... £ 12.00
4 MATTER..... £ 12.00
QMON MONITOR/DEBUGGER ..... £ 18.00
TALENT CARTRIDGE DOCTOR ..... £ 13.00
TASKMASTER ..... £ 25.00

LEISURE

TALENT WEST ..... £ 12.00
TALENT ZKUL ..... £ 12.00
TALENT COSMOS ..... £ 12.00
TALENT THE LOST PHAROH ..... £ 13.00
TALENT JUNGLE EDDI ..... £ 13.00
TALENT NEMESIS..... £ 13.00
TALENT OXFORD TRIVIA ..... £ 16.00
TALENT STRIP POKER ..... £ 13.00
DIGITAL SUPERBACKGAMMON ..... £ 11.00
DIGITAL BLOCKLANDS ..... £ 9.00

BRIDGE PLAYER II ..... £ 17.00
WAR IN THE EAST COMPLETE ..... £ 39.00
WAR IN THE EAST (SCEN. 1) ..... £ 20.00
WAR IN THE EAST (SCEN. 2 OR 3) ..... £ 18.00
PSION CHESS ..... £ 16.00
PSION MATCHPOINT ..... £ 12.00
RUBICON DRAGONHOLD ..... £ 14.00
LEISURE GENIUS SCRABBLE ..... £ 12.00
HYPERDRIVE..... £ 12.00
DIGITAL DROIDZONE ..... £ 9.00
DIGITAL SUPER ARCADIA ..... £ 9.00

OMEGA (DISK 3.5") ..... £ 15.00
OMEGA (CART) ..... £ 14.00
STEVE DAVIS SNOOKER..... £ 12.00
SELLASOFT TANKBUSTERS ..... £ 14.00
TALENT HOVERZONE ..... £ 14.00
STONE RAIDER II ..... £ 13.00
SQUADRONS..... £ 13.00
TOUCH TYPIST ..... £ 12.00
DATALINK 3D SLIME ..... £ 11.00
HEART OF GERN (NEW ADVENTURE) .... £ 15.00
DIGITAL SUPER ASTROLOGER ..... £ 22.00

ALL PRICES INCLUDE VAT & CARRIAGE